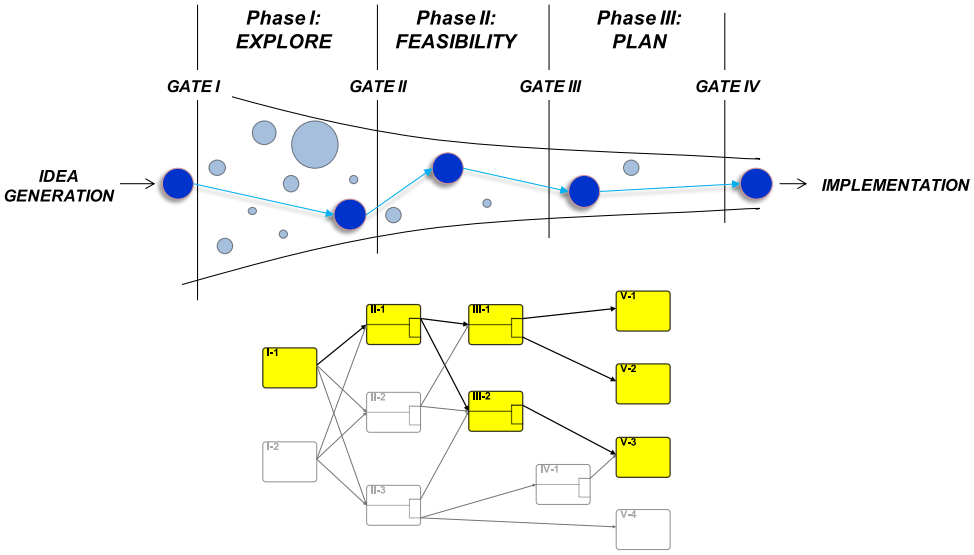
An Integrated Business and Engineering Framework for Synthesis and Design of Processing Networks

Quaglia, Alberto

Publication date:
2013

Document Version
Publisher's PDF, also known as Version of record

An Integrated Business and Engineering Framework for Synthesis and Design of Processing Networks



Alberto Quaglia
Ph.D. Thesis
September 2011

DTU Chemical Engineering
Department of Chemical and Biochemical Engineering

An Integrated Business and Engineering Framework for Synthesis and Design of Processing Networks

Ph.D. thesis

Alberto Quaglia

September, 2013

Computer Aided Process Engineering Center
Department of Chemical and Biochemical Engineering
Technical University of Denmark

CAPEC

DTU Chemical Engineering
Department of Chemical and Biochemical Engineering

Address: Computer Aided Process Engineering Center
Department of Chemical and Biochemical Engineering
Technical University of Denmark
Building 229
DK-2800 Kgs. Lyngby
Denmark
Phone: +45 4525 2800
Fax: +45 4588 4588
Web: www.capec.kt.dtu.dk

Print: J&R Frydenberg A/S
København
October 2013

ISBN: 978-87-93054-08-0

Preface

This dissertation is submitted to the Technical University of Denmark (DTU) in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Ph.D.) in Chemical Engineering. This research project has been running from June 2010 until May 2013. It has primarily been carried out at the Computer Aided Process Engineering Center (CAPEC) at the department of Chemical and Biochemical Engineering of the Technical University of Denmark, in strict and fruitful collaboration with Alfa Laval Copenhagen A/S.

This project has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n 238013.

The project has included a two external research stay, carried out at the Technical University of Eindhoven (The Netherlands) with Assoc. Prof. Edwin Zondervan, and at the University of Maribor (Slovenia), with the group of Professor Zdravko Kravanja. I would like to thank them for their supporting and enriching collaboration.

Moreover, I would like to thank my supervisors Rafiq, Gürkan and Bent, for their support along 3 years of intense, challenging and rewarding work.

I am grateful to my family: my parents Angela and Roberto, and Paola, Pier, Anna and Sara for their support and love.

Here in CAPEC I was so lucky to meet Katrine, and many other wonderful people. Thanks to them all, these years have been inspiring and enriching. If I have accomplished anything professionally and personally I owe most of the credit to all the people I met and I did part of my journey together with, in Italy, Belgium, USA and Denmark. If I look back, I can see that the list of their name is too long to be included in a document, that most of them will not read. I will therefore remember to let them know how important they are, and to try to give them back at least as much as they gave me.

Alberto Quaglia

Kgs. Lyngby, May, 2013

Abstract

The problem of synthesis and design of processing networks corresponds to the generation, evaluation and selection among alternatives with respect to raw materials, process technologies and configurations and product portfolio compositions. This results in a complex and multi-disciplinary problem, in which all the aspects of the problem (technical, economical, regulatory, logistical, etc.) need to be considered simultaneously, in order to be able to identify the optimal design.
Through the developments realized in the last decades, Process Systems Engineering has shown the potential to contribute to this problem, through the development of methods, tools, and solution approaches, under the general framework of Enterprise-Wide Optimization. Despite the level of maturity which these tools have reached and the potential which they have demonstrated, the acceptance of systematic methods and tools for synthesis and design of processing networks in the industrial sector is still lower than what could be expected. One of the key reasons for this lack of acceptance lays in their complexity. The formulation of these problems, in fact, often results in a time-consuming activity, due to the number of data that need to be gathered and of equations that need to be specified. The solution of the optimization problem formulated, moreover, requires expertise in discrete optimization, which is often not part of the standard skills set of design engineers and decision-makers.
This Ph.D. project, therefore, aims at the integration of methods, tools and solution strategies for synthesis and design of processing networks in a computer-aided framework, in order to optimize and facilitate the workflow of problem formulation and solution, allowing simpler, faster and more robust use of such tools. Through the integration of different methods, tools, algorithms and databases, the framework guides the user in dealing with the mathematical complexity of the problems, allowing efficient formulation and solution of large and complex optimization problem.
In this thesis, all developed methods, tools and solution strategies are described, emphasizing their integration in the computer aided framework. The framework is then applied to the formulation and solution of 3 challenging and relevant case studies, highlighting the importance of the tools integration realized in the framework.

Resumé på Dansk

Denne afhandling omhandler udviklingen af en fremgangsmåde til syntese og design af procesnetværk. Dette problem består i at generere, evaluere og udvælge blandt alternativer med hensyn til råvarer, procesteknologier og produkter.
Syntese og design af procesnetværker er et komplekst problem, fordi alle dens aspekter (tekniske, økonomiske, lovgivningsmæssige og logistiske, etc.) må overvejes samtidig, for at være i stand til at bestemme det optimale design.
I de sidste par årtier er problemstillingen vedrørende af procesnetværk blevet undersøgt som en del af *enterprise-wide optimization* (optimering af hele virksomheden); og optimeringsmetoder er blevet udviklet til løsning af syntese- og designproblemet. Anvendelsen af disse nye metoder er stadig yderst begrænset, på grund af den matematiske kompleksitet.
Dette projekt sigter derfor mod at integrere metoder, værktøjer og løsningsstrategier til syntese og design af procesnetværk i et computerunderstøttet *framework*. Dette er med henblik på at optimere og lette arbejdsgangen for problemformulering og løsning, der giver enklere, hurtigere og mere robust anvendelse af så danne værktøjer.
I denne afhandling er alle metoder, værktøjer og løsningsstrategier beskrevet, og deres integration i det computerunderstøttetede framework er præsenteret.
Frameworket er derefter anvendt til formulering og løsning af 3 komplekse og relevante casestudier.
Casestudierne fremhæver betydningen af integrationen af værktøjerne, hvilket realiseres ved hjælp af frameworket. Som følge af integrationen er formuleringen og løsningen af casestudierne forenklet.

Contents

Part I

Introduction

1

Introduction

1.1 Introduction to chemical processing networks

In general terms, a chemical company is an enterprise which operates the business of transforming raw materials into value-added products, through a series of physical and chemical operations, which constitutes a chemical process.
Chemical processes can have different levels of complexity, spanning from simple cases in which the transformation is constituted by a single processing step, to more complex cases, in which several transformation steps are operated in an integrated manner, so that the product of a process step becomes the raw material for the next one. In the latter case, each individual process becomes a node of a complex *processing network*, through which the transformation of several resources and raw materials into different products is realized. An example of a processing network is shown in figure 1.1.
For multinational companies, processing networks can integrate a large number of processes located in different geographical locations, and complex material flow patterns may exist. Notable examples include, among others, BASF's *Verbund* sites, in which a large number of processes are operated in an integrated manner (Bruggemann *et al.*, 2008).
For a chemical company, a processing network represents a key asset, since it determines the ability of the company to perform its core activity, which is the production of value-added products. Consequently, the design of a new processing network (as well as of a major modification of an existing one) is a task of major importance, which have a crucial impact on every chemical company.

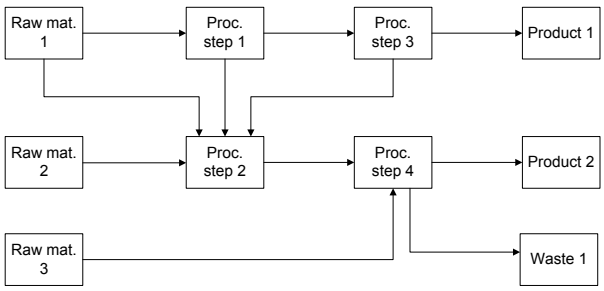


Figure 1.1. Illustration of a processing network

1.1.1 Construction and modification of processing networks

Processing networks are constantly subject to modification, because of the dynamic changes occurring in the business environment (Kurkkio *et al.*, 2011). In particular, alterations of the processing network structure may occur due to:

- Capacity expansion/reduction
- Process development
- Supply chain optimization
- Product portfolio modification
- Raw materials modification

Capacity expansions/reductions aim at modifying the productivity of one or more products, in order to respond to changes in market size and market shares.
Process developments focus on the improvement of the production process, through optimization of the operating conditions or modification of the process technology employed. Usual drivers for process developments include improvement of product quality or yield, as well as reduction of processing cost or environmental footprint. In some cases project developments can also be motivated by the emergence of disruptive technologies, new regulations or market trends which require the transition to a different process technology.
Supply chain optimizations focus on the material flows through the network, aiming at improvement from a cost/benefit perspective, including optimization of inventory levels, planning, scheduling, centralization/ decentralization of storage and production sites, etc.
Product portfolio modifications are concerned with the introduction of new products and/or the discontinuation of existing ones, based on consumer needs and market trends, as well as on the company strategy.
Raw materials modifications aim at modifying the network in order to adapt to changes in the supply of one or more raw materials, with respect to both quality and quantity.
Any major modification of a processing network requires the construction of a processing plant or of a part of it. In most cases, this is a complicated and expensive activity, which in general is associated to long-lasting capital-intensive projects. In order to design and build a large production plant, in fact, a massive capital investment is needed to sustain the activity of process design and detailed engineering, the purchasing of all necessary equipment, and all cost associated to plant construction, commissioning and start-up. Moreover, the completion of each of these tasks requires a considerable amount of time, and consequently the overall project may span over several years.
In addition, the construction of a chemical processing plant has an environmental impact, related to the land covered by the plant footprint, as well as the emissions for production, transportation and assembly of the equipment. Furthermore, a certain amount of resources is also committed by the community in which the plant is located, and public investments are done to favour the construction of the new plant, for example by ensuring services to plant workers (housing, schooling, etc.)

or building infrastructure for transportation or telecommunication.
In order for all these investments to be fruitful, the chemical processing plant which is built needs to be able to operate in a profitable way for a sufficient time, generating revenues to achieve at least the minimum expected return on investment, hence paying back the investment committed by all the stakeholders involved. As a consequence, the decision-making problem associated to the synthesis and design of a processing network is of crucial importance for the chemical company involved, as well as for the human society to which such a company contributes.

1.1.2 Design of processing networks: a complex and multidisciplinary problem

The optimal design of a processing network from an enterprise-wide perspective is a complex decision-making problem, which requires the integration of a number of different disciplines and knowledge. In fact, while chemical engineering knowledge remains the key competence required to design a process (given raw materials and products specifications), many other considerations and expertises are needed when the design problem is extended to the enterprise-wide level, incorporating decisions such as the selection of raw materials and products, the geographical location of production plants, etc.
In particular:

- Marketing knowledge is needed in order to estimate market prices and volumes for different products and different product specifications, as well as their trends over time.
- Supply chain knowledge is required to select the raw material source, in order to ensure a stable supply and good pricing.
- Regulatory expertise is required in order to ensure that both process and products comply with all relevant regulations, with respect to Environmental Health and Safety (EHS) and product quality.
- Intellectual Property (IP) expertise is needed in order to evaluate whether the proposed network violates any existing patents, and in that case to establish a strategy to ensure *freedom-to-operate* (for example through licensing).
- Financial expertise is required in order to optimize the investment from a cash-flow point of view, and elaborate strategies to obtain the credit needed for it.

It is important to reflect on the fact that these different aspects are not independent, but interdependent. For example, the design of the process requires the definition of raw materials and product specification. At the same time, the selection of a given product requires the knowledge of its market value and of its production cost, which is a function of the design of the process which has been selected.
Therefore, when seeking to determine the optimal processing network design from an enterprise-wide perspective, all aspects of the problem have to be considered simultaneously in the decision-making problem. Consequently, the problem becomes complex and multidisciplinary.

1.1.3 Uncertainty

As underlined above, the synthesis and design of a processing network is based on a number of multidisciplinary data and premises. Most of these aim at describing events which will realize in the future. Consequently, at the time at which design decisions are taken, only hypothesis can be made on the value of these data, based upon past and current observations (e.g. market fluctuations of a product). These data are therefore subjected to a certain degree of uncertainty.
Uncertainty needs to be carefully managed at the design phase, through the implementation of strategies aiming at minimizing it or at mitigating its potential negative consequences on the performances of the processing network.
Minimization of data uncertainty can be achieved by further analysis and investigation of the uncertain phenomena, in order to accumulate more information and achieve a "more certain" knowledge about them. Modelling studies, lab or pilot-scale experiments are examples of investigations aiming at reducing the technical uncertainty, while marketing studies are related to market uncertainty.
Mitigation strategies, on the other hand, aim at reducing the impact that a negative realization of those uncertain data may have on the performances of the processing network. Safety factors used in equipment and process design are examples of mitigation strategies used to manage technical uncertainty. Long-term agreements with suppliers and customers, aiming at defining prices and volumes are examples of mitigation strategy for market uncertainties.
In both cases, the protection against negative consequences related to uncertainty is obtained at a price. Extended investigations of uncertain phenomena, in fact, are in general resource-intensive, and delay the overall project. Similarly, uncertainty mitigation strategies are in general associated to additional cost or performances decay. As a consequence, in most of cases practical reasons suggest not implementing these strategies for every source of uncertainty, but instead directing them on those, which have a significant impact on the performances of the overall network.
Depending on the case, the dominating source of uncertainty may be associated with different data. For example, if the design is related to a new product, the estimation of its market price and volume is likely to be subjected to a large uncertainty. Likewise, if a new process technology is considered, its performance data are likely to be a non-negligible source of uncertainty.

1.1.4 Industrial practice: iterative solution

Processing industries are in general organized in a functional structure, with separate departments working in a coordinated and integrated manner on enterprise-wide projects and activities. As pointed out in section 1.1, an important example of enterprise-wide activity is the design and retrofit of processing networks.
As explained in section 1.1.2, this requires different knowledge and expertise in order to shape the core activity of the company by making strategic decisions (e.g. the selection of raw materials and product portfolio) and tactical decisions (e.g. the determination of operating conditions for the process and of material flows through the network).

In industrial practice, this problem is usually tackled by business-oriented and engineering-oriented departments, which work in a coordinated manner on the strategic and tactical aspects of the synthesis and design problem. Business-oriented departments deal primarily with strategic decisions, which are taken on the basis of marketing and financial considerations, employing tools or indicators such as Balanced Scorecard and project NPV. On the other hand, engineering-oriented departments focus on tactical decisions, related to design and optimization of processes with the help of tools such as process simulators (Quaglia *et al.*, 2012).
Because of the interdependencies between the different decision-making layers (strategic and tactical) explained above, a sequential solution of the two layers (e.g. fixing the strategy first, and then define a tactic for it) is often not possible, and the two aspects of the problem need to be solved iteratively. A schematic representation of the required workflow is given in figure 1.2.

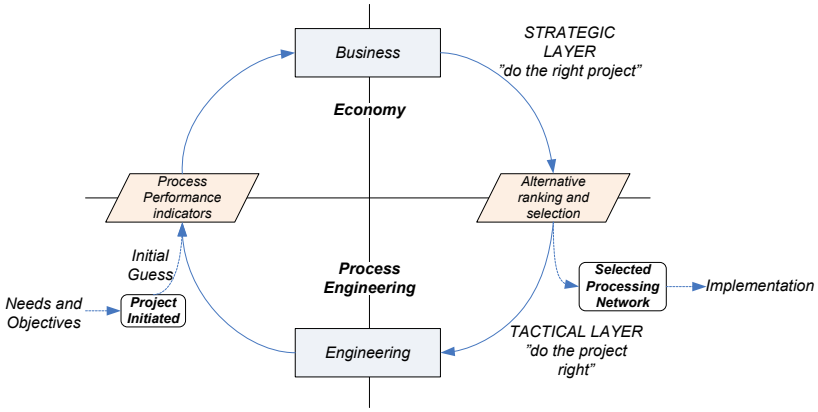


Figure 1.2. Schematic representation of iterative solution of enterprise-wide decision-making problems (Quaglia *et al.*, 2012)

The uncertainty involved in the decision-making problem is in general tackled through the evaluation of the performances of the given network under a "worst", "average" and "best" case scenario, defined with respect to the value of the uncertain data.
Although widely adopted, this approach presents some drawbacks. Because of its complexity and multidisciplinary nature, the design procedure cannot be automated, and its execution may become time consuming and cumbersome, especially when large problems are considered. As a consequence, in the design procedure it is often possible to evaluate only a limited number of alternatives. Moreover, analysis techniques such as sensitivity analysis and scenario planning are in general applied to each of the layers separately, and therefore are not able to capture the interdependencies between the business and engineering aspects of the problem.
In order to partially overcome these drawbacks, in many cases a second level of decomposition is employed, based on the so called *development funnel* approach (see figure 1.3). The basic idea of the development funnel is the decomposition of the problem into a sequence of design sub-problems, which are solved with an incremental level of accuracy (Clark, 1993). In the first phase of the funnel, a simplified version of the design problem is solved, disregarding part of its complexity. Because

of the simplifications, limited time and resources are needed to complete the task, and therefore a wider range of alternative designs can be explored to identify the optimal one. On the other hand, because of the simplifications, an approximated solution is obtained, whose performances are therefore subject to high level of uncertainty.
The solution of the first phase is evaluated in a gate decision point. If the performances meet the expectations, the design project is transferred to the second phase. In this second phase, the design is performed again in a more detailed version, evaluating the alternatives which appeared of most interest in the previous phase; a more detailed design is obtained and evaluated in a second gate decision. The procedure is repeated until the project is either discarded at a gate point because no satisfactory design can be found, or when the full complexity of the problem has been considered (Wheelwright, 1992).

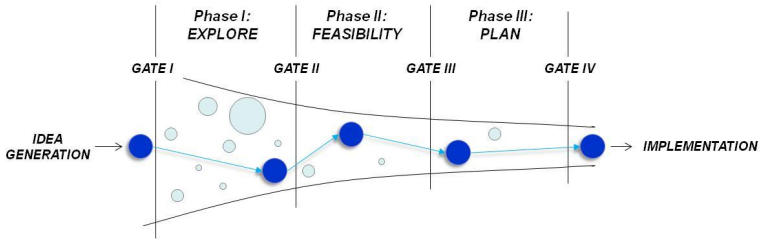


Figure 1.3. Development funnel and stage-gate project management

1.2 Systematic synthesis and design of processing networks

In this section, a formal definition of the problem of synthesis and design of processing networks is given, and the approaches developed in the scientific literature in order to obtain a systematic solution of this problem are discussed.

1.2.1 Problem definition

The problem of synthesis and design of processing networks in its general form can be stated as follows (Quaglia *et al.*, 2012):
Given:

1. *a supply and demand equilibrium between different market goods*
2. *process technologies that allow the transformation of a set of low value goods (raw materials) in a mixed set of:*
   (a) *high value goods (products),*
   (b) *medium-low value goods (by-products),*
   (c) *cost items (wastes)*

at the expense of:

(d) operational costs (OPEX)

(e) capital investment (CAPEX)

3. *a mathematical model describing each of the processes in the network*

4. *a quantitative definition of optimality (objective function), as a scalar function of process parameters through a number of fixed parameters (prices or weights).*

The objective is the selection of:

1. *the subset of raw materials, product and by-products*

2. *the processes topology*

and the calculation of:

3. *the material flow through the processing path*

4. *the value of all the optimization variables*

Which maximize the objective function.
Furthermore, for the identified optimal conditions, the following is calculated:

1. *sizing parameters and utilities and chemicals consumption for each of the selected processes*

2. *process, sustainability and economical indicators (i.e. products yields, LCA etc.).*

From this general formulation, specific problems can be generated by different variable specification, such as:

- Resources allocation problem: by fixing the raw material selection and incoming flow.
- Supply chain problem: by fixing the product selection and material flow.
- Strategic planning problem: by fixing the processing network topology.
- Retrofit problem: by fixing all or part of the processing network topology and raw material and product selection and flows.

1.2.2 Systematic methods for synthesis and design of processing networks

Because of the relevance of the problem, in recent years many authors have focused on the development of systematic methods for synthesis and design of processing networks. Through their work, these researchers aimed at developing methods, tools and formal approaches allowing designers to identify better design, while at the same time reducing the time and resources needed for the design problem. Depending on the type of methodology developed, these systematic approaches can be classified into 4 main categories:

- evolutionary modifications
- insight-based methods
- mathematical programming
- hybrid methods

In the next sections, these approaches will be briefly described.

Evolutionary modification

Evolutionary modification is a design strategy consisting in devising minor modifications to an existing design, in order to obtain an improvement in its performances. The modification procedure is iterated, producing several design generations exhibiting progressively improved characteristics. The procedure is stopped when no further improvement can be obtained, or when the outcome is considered satisfactory, or after a fixed number of generations.
In order to perform design through evolutionary modification, algorithms, methods and tools are needed to:

1. generate an initial design (if not available)
2. identify which parts of the design are less promising, and should therefore be modified
3. suggest a modification which can improve the characteristic of the design object

The development of such methods and tools have been the focus of many authors in the scientific literature, starting from the early work of King and Barnes (1972) who developed an heuristic-based approach to flowsheet design via evolutionary modification, and applied it to 2 case studies obtaining a local solution.
McGalliard and Westerberg (1972) developed an evolutionary algorithm for flowsheet design, based on the application of structural sensitivity analysis to quickly screening among modified design.
Stephanopoulos and Westerberg (1976) studied and systematized the logic used in evolutionary modification, and applied the method to the synthesis of a separation process.
More recently, Halim and Srinivasan (2002) developed an insight-based methodology for evolutionary modification of process design, aiming at waste minimization. Carvalho *et al.* (2008) proposed a sensitivity-based methodology for the design of sustainable processes, based mass and energy indicators, and implemented it in the software package SustainPro (Carvalho *et al.*, 2013).

Insight-based methods

Insight-based methods are based on the hierarchical decomposition of the design problem into sub-problems, such as reaction systems, separation systems, etc.

In each sub-problem, design decisions are taken based on insights, obtained from heuristics (based on design experience) or engineering knowledge (such as physical properties or thermodynamics) (Barnicki and Siirola, 2004).
Because of the interconnections existing between these sub-problems (Douglas, 1988), an iterative procedure is required for the solution of the overall design problem.
Siirola (1996) and Barnicki and Siirola (2004) developed and applied a task-oriented hierarchical systematic generation approach, based on an artificial intelligence planning paradigms such as *means-ends analysis* (Simon, 1973), and applied it successfully to the synthesis and design of processes at the Eastman chemical company.
Many contributions have been focusing on the use of insight-based methods for the design of separation-based systems (Barnicki and Fair, 1990; Brunet and Liu, 1993). Among these studies, Bek-Pedersen and Gani (2004) focused on the design of distillation-based separation systems based on the driving-force concept.

Mathematical programming

Mathematical programming approaches, developed under the general framework of enterprise-wide optimization (Grossmann, 2005), are based on the decomposition of the synthesis and design problem in three steps. The first step consists of the representation of the solution space for the design problem by means of a superstructure, containing a finite number of processing units together with their possible interconnections. The second step consists of the formulation of the design problem as a mathematical optimization. This requires the definition of an objective function indicating the goal of the synthesis and design problem, as well as models for each of the elements of the superstructure. As a result, the design problem is cast as a Mixed Integer Non Linear Program (MINLP), in which binary variables are used to represent boolean decisions. In the third step the problem is solved, in order to determine (within the search space defined by the superstructure) the optimal design, according to the criteria defined by the objective function.
The main feature of optimization-based approaches is the possibility of screening among a large number of design alternatives, through a procedure which simultaneously considers all the aspects of the design problem, as well as the interdependencies existing between them. This powerful feature comes at the price of an increased mathematical complexity.
In most of cases, these methods result in the formulation of large and complex Mixed Integer Non Linear Programming (MINLP) problems, and considerable computational resources are needed to obtain the solution (Duran and Grossmann, 1986b).
Over the last decades, many authors focused on the development of solution algorithms for such problems, based on generalized bender decomposition (Floudas, 1995), outer approximation (Duran and Grossmann, 1986a), extended cutting plane methods (Westerlund, 1998) and branch-and-bound (Gupta and Ravindran, 1985).
In particular, the complexity of the mathematical problem is driven by i) problem size, especially with respect to number of discrete variables, ii) constraints complexity, with respect to non-linearity and non-convexity and iii) number of uncertain data.
Many contributions have been made that overcome these challenges. Turkay and Grossmann (1996) introduced a logic-based framework for problem formulation

known as Generalized Disjunctive Programming (GDP). The main feature of GDP is to employ big-M or convex hull reformulation to formulate mathematically the logic of the design decisions.
Similarly, reformulations of complex (non-linear, non-convex) constraints have been developed, in order to provide an approximated solution or a under/overestimator to be employed in bi-level decomposition schemes. Examples include piecewise linearization of bilinear and non-linear terms (Bergamini *et al.*, 2005, 2008; Karuppiah and Grossmann, 2006; Bogataj and Kravanja, 2012), as well as non-linear reformulation of non-convex terms (Karuppiah and Grossmann, 2006; Bogataj and Kravanja, 2012).
With respect to data uncertainty, among the different approaches which have been developed, the dominant methods appearing in the literature are:

- optimization under uncertainty
- optimization with flexibility
- multi-parametric programming

Optimization under uncertainty methods are based on the reformulation of the problem as a two stage stochastic programming (Birge and Louveaux, 1999), which is solved in order to identify a design which is feasible over the whole uncertainty domain, and whose expected performance is optimal.
Optimization with flexibility, on the other hand, aims at identifying solutions which are optimal for the expected realization of the uncertain data, and at the same time have the flexibility to remain feasible over the entire uncertainty domain (Pistikopoulos and Mazzuchi, 1990; Straub, 1993), or over a fraction of it (Swaney and Grossmann, 1985a,b).
Multi-parametric programming exploits the sensitivity information contained in the Lagrange multiplier to provide an explicit map of the optimal solution as function of the uncertain data (Dua and Pistikopoulos, 1998; Pistikopoulos, 2007; Dominguez and Pistikopoulos, 2010).

1.2.3 Hybrid methods

Alongside the above described approaches, some authors focused on the development of hybrid methods, based on the integration of insight-based methods and mathematical programming, in which the insight-based logic is used to reduce the computational burden for the mathematical optimization.
As an example, Jaksland *et al.* (1995), Hostrup *et al.* (2001), Tay *et al.* (2011) integrated thermodynamic insights and mathematical optimization methods for the synthesis and design of optimal flowsheets for separation of multicomponent mixtures and biorefinery. Similarly, Barnicki and Siirola (2004) proposed an approach in which means-ends analysis is used in order to generate a superstructure, which is then treated with mathematical optimization to obtain the optimal flowsheet. Moreover, Lutze *et al.* (2013) developed a phenomena-based framework for the synthesis and design of process flowsheets including process intensification options, in which engineering insights are used in order to identify the desirable phenomena, and generate the search space for the mathematical optimization.

1.2.4 Use of systematic design methods in industrial practice

As explained above, evolutionary modification and insight-based methods cannot capture the interdependencies between the different sub-problems in which the design problem is decomposed. Moreover, these methods focus primarily on the engineering layer of the decision-making problem, and do not integrate the business-related aspects.
On the contrary, mathematical programming has the capability to consider simultaneously all the aspect of the decision-making problem, and therefore it represents the most suitable approach for the formulation and solution of enterprise-wide problems, such as the synthesis and design of processing networks. In particular, the ability of this class of methods to cope with extremely large numbers of alternatives makes them extremely suitable for the early stages of the development funnel (figure 1.3). Nevertheless, despite these powerful features and the level of maturity reached through the contribution of many authors in the scientific literature, the penetration of optimization-based design methods for the synthesis and design of processing networks in the industrial practice is somewhat less than what might have been expected (Barnicki and Siirola, 2004), and a gap still seems to exist between the features and claims of these methods and the requirements of professionals and industrial practitioners.
While to some extent the cause of this acceptance gap can be identified in the natural resistance that large organizations offer against the adoption of new *modus operandi*, it should be acknowledged that the complexity of such methods represents an obvious limiting factor for their adoption in industry, since large amount of information, knowledge and specialized expertises are needed for their use.
The formulation of the design problem, in fact, requires managing large amount of information, in the form of data and models. Being the problem multidisciplinary, this knowledge is referred to different disciplines, and requires the use of different sources and databases. Consequently, data collection and systematization requires a considerable investment of time and resources. Moreover, the mathematical resolution of the problem is often not trivial, and may require specialized skills and expertise with handling such problems in order to select and implement different solution strategies, depending on the specific problem which is considered.

1.3 Objective of the Ph.D. project

The objective of this Ph.D. project is to develop a method for the solution of the problem of synthesis and design of processing networks. To meet this objective, a systematic workflow for problem formulation and solution is to be developed, together with all the needed models, methods, tools and solution strategies. Furthermore, all these components need to be integrated in a computer-aided framework for the execution of the systematic workflow. Finally, application of the workflow together with the computer-aided framework is to be highlighted through the formulation of three case studies.

1.4 Structure the Ph.D. thesis

This thesis is organized in 4 parts, divided in 11 chapters.
Part I consists of this chapter, in which an *introduction* to this Ph.D. project is given.
In **Part II**, the integrated business and engineering framework is presented, through the introduction of its main components. In particular, the *key concepts* on which the proposed framework is based are defined in chapter 2. The key components of the framework are then introduced, through the presentation of the *mathematical models* in chapter 3, of the *workflow* for problem formulation and solution in chapter 4 and of the *solution strategies* integrated in the framework in chapter 5.
All *supporting methods and tools* which have been developed in order to facilitate the formulation and solution of the design problem are described in chapter 6, and their practical implementation in a *software* tool is presented in chapter 7.
In **Part III**, the key features of the framework are highlighted through the solution of *three case studies*. The first case study is constituted by the *Numerical Benchmarking Problem* (NBP), proposed for demonstration and benchmarking purposes, presented in chapter 8. The second case study focuses on *soybean processing*, resulting in the formulation of a large synthesis and design problem described in chapter 9. The last one is related to the design of a *wastewater treatment and reuse* network for oil refinery wastewater, presented in chapter 10.
Part IV highlights the *conclusions* from the Ph.D. project and a perspective on future developments of the framework, presented in chapter 11.

1.5 Dissemination of the results obtained in this project and presented in this thesis

Along the duration of this Ph.D. project, the concepts and methods developed and the results obtained have been presented and discussed in 13 international conference presentations. Furthermore, the great majority of the methods, tools and results developed and obtained have been disseminated through publication in scientific journals. In particular, in Quaglia *et al.* (2012) the key concepts and models presented in chapter 2 and 3 have been presented. The workflow for formulation and solution under uncertainty (chapter 4), the solution methods for the stochastic problems (chapter 5) and their applications to the first two case studies have been described in Quaglia *et al.* (2013a). In Quaglia *et al.* (2013b), the data management structure (chapter 6) is described, and its role in supporting and facilitating the workflow of problem formulation is highlighted. Finally, in the recently submitted Quaglia *et al.* (submitted) the strategies for the solution of non-convex problems presented in chapter 5 are presented, and their application in solving large scale multi-scale problems such as the oil refinery wastewater treatment and reuse (chapter 10) are demonstrated.

Part II

The Business and Engineering Framework

2

Definition of key concepts

The developed framework for synthesis and design of processing networks employs an ontological representation of processing network structures. In this chapter, the key elements constituting this representation are defined.

2.1 Superstructure, process steps and process intervals

As explained in the previous chapter, in mathematical programming a superstructure representation of the search space for the design problem is employed. Through the superstructure, all possible alternatives with respect to the topology of the processing network are represented based on elementary graph theory, in which raw material options are represented as inputs, product alternatives as outputs, processing elements as nodes and connections between processing elements as arcs (Floudas, 1995). The developed framework for synthesis and design of processing networks is based on a stage-wise superstructure representation, as the one shown in figure 2.1. In the stage-wise representation, raw material alternatives are represented as first column of the superstructure, while product alternatives constitute the last column. The overall process, therefore, proceeds from the left to the right hand-side of the superstructure.

The process to convert raw materials into products is represented as a sequence of *process steps*, which are represented as columns of the superstructure. As the name indicates, each process step constitutes a step in the transformation of raw materials into products. An example of process step is the removal of a given contaminant from a process stream. Each process step contains one or more *process intervals* (represented as boxes in the superstructure). A process interval is defined as a technological alternative for the execution of a process step. Examples of process intervals for the above mentioned contaminant removal step may include separation via distillation, via extraction, via selective conversion of the contaminant, etc. Finally, possible material flows through the network of process intervals are represented by connections. These include forward connections (in which the flow is in the same direction as the overall processing flow), recycles and backward connections (when the flow is opposite to the overall processing flow).

The structure of above described stage-wise superstructure representation presents a strong analogy with the stage-wise approach developed for heat exchanger networks design (Yee *et al.*, 1990). With respect to the approach developed by Yee, in this representation incoming and outgoing hot streams are substituted by raw materials and products, temperature intervals by process step, and heat exchangers by process

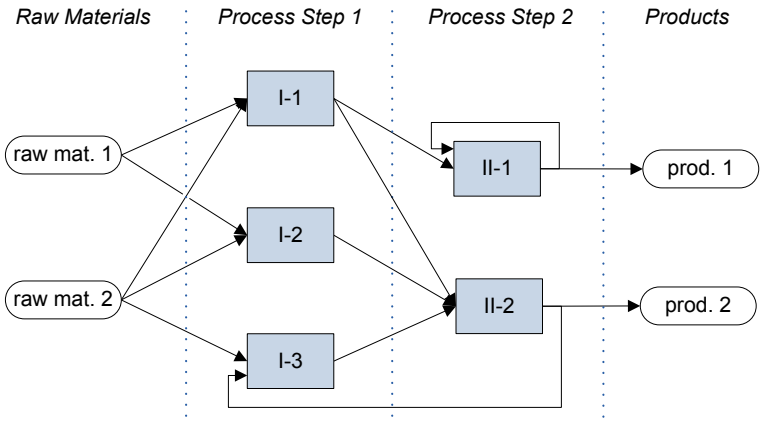


Figure 2.1. A superstructure for processing network synthesis and design

intervals.

2.2 Elementary process tasks

As described in the previous section, a process interval is a segment of a process, through which a process step is performed. Depending on the step and of the process technology employed, process intervals can be characterized by process flowsheets of different structure. Moreover, a process interval may contain multiple unit operations, and have multiple material input and output.
Within the developed framework, the complexity related to the structural variability of process intervals is managed through a functional representation, based on process tasks. *Process tasks* are functional descriptors, through which the transformations occurring to a stream in a process are described in a generic manner (Zondervan *et al.*, 2011). Within the developed framework, six process tasks are employed to describe the variety of transformations occurring in a chemical process. These tasks are flow mixing, utility dosage, reaction, waste separation, separation and stream divider (figure 2.2).

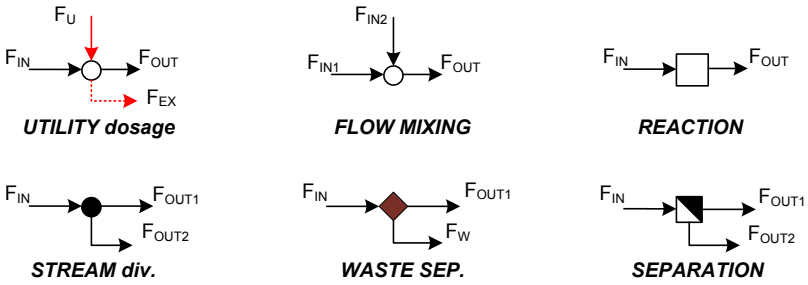


Figure 2.2. Elementary process tasks

The process tasks are defined as follows:

- Flow mixing: represents the task of mixing of two or more process streams
- Utility dosage: represents the consumption of utilities or chemicals, either mixed together with the process stream (e.g. chemicals) or kept separate (e.g. electricity)
- Reaction: represents the occurring of a chemical reaction changing the composition of the process stream
- Waste separation: represents the separation of a waste stream from the process stream
- Separation: represents the separation of the process stream into two streams of different composition
- Stream divider: represents the separation of the process stream into two streams of equal composition

The above described process tasks constitute the building blocks, which can be combined in different configuration, to obtain a description of the process segment represented by a process interval, in a generic manner.

2.3 Generic process interval model

A simple and generic representation of a process interval, based on the elementary process tasks described above, is constituted by the generic process interval model, represented in figure 2.3.

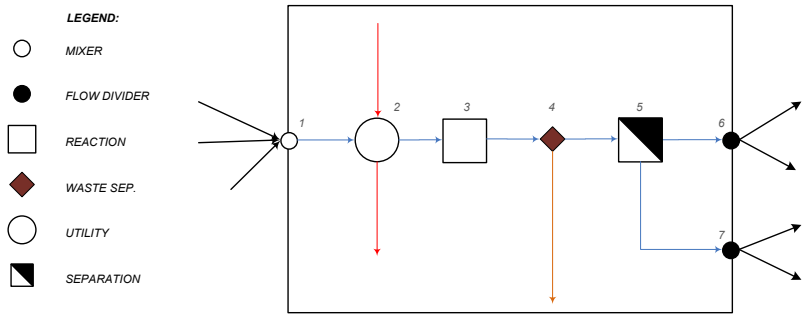


Figure 2.3. Generic process interval model structure, represented as a sequence of process tasks

In the generic process model, the chemical process performed within a process interval is represented. All incoming flows are mixed, utilities and chemicals are then applied to the process stream before feeding it to the reaction task. After the reaction, a waste stream is separated and the process stream is fed to a separation task. Both primary and secondary outlets of the separation are fed to stream dividers,

whose different outlets are fed to different process intervals. The structure allows the possibility of bypassing one or more of the above mentioned task, in case those are not executed in the process. The generic process interval model allows the representation of a wide variety of processes through a common structure, hence greatly simplifying the task of modeling and data handling.

3

Models

In this chapter, all mathematical models adopted for the formulation of the synthesis and design problem are reported. These include the models for each of the process tasks defined in the previous chapter, the generic process interval model, the model of the superstructure, the economical indicators and the objective function models.

3.1 Mathematical notation

In an attempt to simplify the understanding of the mathematical models and equations reported in this thesis, a standard mathematical notation has been adopted. Equations are reported in matrix form, where index i represents the chemical species, k the process intervals (from which streams are originated), kk the process intervals (to which streams are fed) and rr the chemical reactions. In general, constant terms (data or parameters) are represented by greek letters, while latin alphabet is used to describe optimization variables. Exception to this notation is constituted by the molecular weight, for which the standard notation in latin alphabet (MW) is maintained. All flow variables F indicate mass flows.
A complete list containing all symbols, indexes and operators employed in the thesis is reported in the appendix.

3.2 Process task models

In this section, the linear models employed to describe each of the process tasks presented in figure 2.2 are introduced.

3.2.1 Flow mixing

The task of mixing f flows is calculated from the material balance as:

$$F^i = \sum_f (F^i_{0,f}) \tag{3.1}$$

where the index i indicates the chemical species and f incoming streams; the variable $F^i_{0,f}$ represents the mass flow of component i in the incoming stream f, while F^i is referred to the outlet mixed stream.

3.2.2 Utility dosage

The flow of utility and chemicals F^{ut}_U required in the utility task is calculated as:

$$F^{ut}_U = \mu^{ut,i} \cdot F^i_0 \tag{3.2}$$

where $\mu^{ut,i}$ is the specific consumption of utilities and chemicals ut with respect to incoming flow of component i. The utility task model is given as:

$$F^i = \sum_f (F^i_{0,f}) + \alpha^{i,kk} \cdot F^{ut}_U \tag{3.3}$$

Where $\alpha^{i,kk}$ is the fraction of chemical or utility mixed with the process stream.

3.2.3 Reaction

The reaction task is modeled through a stoichimetric reaction model as:

$$F^i = F^i_0 + \sum_{rr,react} (\gamma^{i,rr} \cdot \theta^{react,rr} \cdot F^{react}_0 \cdot \frac{MW^i}{MW^{react}}) \tag{3.4}$$

where rr is the index identifying the reaction and $react$ is the element of i which represents the key reactant for the reaction; $\gamma^{i,rr}$ is the molar stoichiometry for reaction rr, $\theta^{react,rr}$ is the conversion with respect to the key reagent $react$ and MW is the molecular weight.

3.2.4 Waste Separation

The waste separation task model is given as:

$$F^i = F^i_0 \cdot (1 - \delta^i) \tag{3.5}$$

where δ^i is the fraction of component i which is separated in the waste flow. Similarly, the separated waste flow F^i_W is calculated as:

$$F^i_W = F^i_0 \cdot \delta^i \tag{3.6}$$

3.2.5 Separation

The primary outlet from a separation task is given as:

$$F^i = F^i_0 \cdot \sigma^i \tag{3.7}$$

while the secondary flow F^i_S is:

$$F^i_S = F^i_0 \cdot (1 - \sigma^i) \tag{3.8}$$

where σ^i is the split factor for separation of component i.

3.2.6 Flow divider

The primary outlet from a stream divider is calculated as:

$$F^i = F^i_0 \cdot \omega \tag{3.9}$$

while the secondary F^i_S is:

$$F^i_S = F^i_0 \cdot (1 - \omega) \tag{3.10}$$

where ω is the fraction of incoming flow which is sent to the primary outlet.
In cases where the divider fraction is not specified as data it is considered an optimization variable. In such cases, the model becomes:

$$F_f^i = F_0^i \cdot SM_f \tag{3.11}$$

$$\sum_f (SM_f) = 1 \tag{3.12}$$

Where SM_f is the fraction of incoming flow which is sent to the outlet f

3.3 Generic process interval model

Employing the process task models presented in equations 3.1 - 3.12, the generic process interval model presented in chapter 2.3 is obtained as equations 3.13 - 3.20, where the internal flows are named according to figure 3.1.

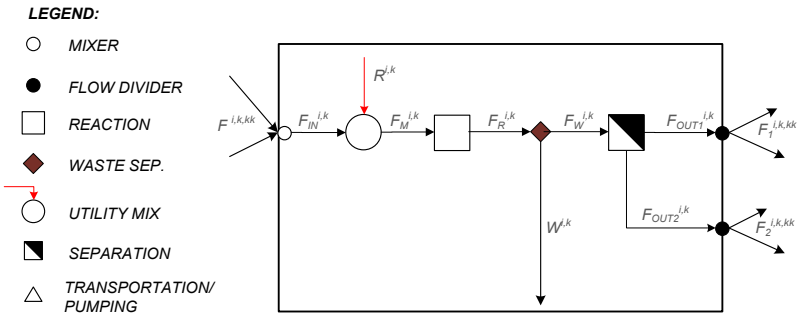


Figure 3.1. Process Interval model structure. Highlighted, the internal flow variables

$$F_{IN}^{i,kk} = \sum_k \left(F^{i,k,kk}\right) \tag{3.13}$$

$$R^{i,k} = \mu^{ut,i,k} \cdot F_{IN}^{i,k} \tag{3.14}$$

$$F_M^{i,k} = F_{IN}^{i,k} + R^{i,k} \cdot \alpha^{i,k} \tag{3.15}$$

$$F_R^{i,k} = F_M^{i,k} + \sum_{rr,react} \left(\gamma^{i,rr,k} \cdot \theta^{react,rr,k} \cdot F_M^{react,k} \cdot \frac{MW^i}{MW^{react}}\right) \tag{3.16}$$

$$F_W^{i,k} = F_R^{i,k} \cdot (1 - \delta^{i,k}) \tag{3.17}$$

$$W^{i,k} = F_R^{i,k} - F_W^{i,k} \tag{3.18}$$

$$F_{OUT1}^{i,k} = F_W^{i,k} \cdot \sigma^{i,k} \tag{3.19}$$

$$F_{OUT2}^{i,k} = F_W^{i,k} - F_{OUT1}^{i,k} \tag{3.20}$$

The formulation of the generic process interval model for an example process is described in appendix B.

3.4 Superstructure model

The superstructure model is composed of two parts: the logic model, which describes the logic related to the selection of process intervals, and the flow model, which describes the material flow through the network of process intervals.

3.4.1 Logic model

The logic model is consists of logical and activation constraints.

Logical constraints

Logical constraints are equations based on the binary variables representing the selection of process intervals, and are used to eliminate undesired or infeasible solutions from the search space. From a design perspective, logical constraints represent the mathematical formulation of all conditions that the structure of a processing network should satisfy, in order to be a feasible solution of the design problem. These conditions may reflect engineering knowledge (e.g. with respect to process technology), commercial insights (e.g. with respect to product portfolio) or regulatory considerations (e.g. with respect to EHS or *freedom-to-operate*).
Examples of logical constraints are given in table 3.1, where y^k is a binary variable which is equal to 1 if interval k is selected, and zero otherwise. Other logical conditions can be formulated as constraints through the use of propositional logic (Raman and Grossmann, 1991).

Table 3.1. Examples of formulation of logical constraints through propositional logic

Condition	Proposition	Logical constraint
select at least one interval between a, b and c	$y^a \vee y^b \vee y^c$	$y^a + y^b + y^c \geq 1$
select a and at least one between b and c	$y^a \wedge (y^b \vee y^c)$	$y^a = 1;\ y^b + y^c \geq 1$
select one between a, b and c	$y^a \oplus y^b \oplus y^c$	$y^a + y^b + y^c = 1$
if a is selected, b should also be selected	$y^a \Rightarrow y^b$	$y^a - y^b \leq 0$

Activation constraints

Activation constraints are used to express relations between binary and continuous variables, hence providing a model to describe the consequence of the decisions represented by the binary variables (Floudas, 1995).
An example of an activation constraint is given as:

$$y^k \cdot x^k_{LO} \leq x^k \leq y^k \cdot x^k_{UP} \tag{3.21}$$

Through this constraint, variable x^k is bounded by x^k_{LO} and x^k_{UP} if the corresponding interval k is selected ($y^k = 1$). Otherwise, the variable x^k is set to zero.

3.4.2 Flow model

The flow model describes the material flows through the network of process intervals. Depending on the problem specification with respect to the selection of process

intervals within a process step, the flow model may be defined as *multi-stream* or *single-stream*.

Multi-stream flow model

The multi-stream flow model represents the most general formulation of the flow model, and corresponds to the case in which more than one interval can be selected for each process step. In this case, a stream is divided and multiple sub-streams are originated, hence the name "multi-stream".
The multi-stream flow model is given as:

$$F_f^{i,k,kk} = \sum_f (F_{OUT,f}^{i,k} \cdot SM_f^{k,kk} \cdot \zeta_f^{k,kk}) \tag{3.22}$$

Where $F_f^{i,k,kk}$ is the flow of component i between treatment k and treatment kk; $\zeta_f^{k,kk}$ is a parameter defining the connections existing in the superstructure, which is 1 if a connection exists between the outlet f of interval k and the inlet of interval kk, and is zero otherwise; $SM_f^{k,kk}$ is the fractions of outlet flow f from interval k which is fed to interval kk (split factor), which is subject to the consistency condition:

$$\sum_{kk} (SM_f^{k,kk}) = y_k \quad \forall f;\ \forall k \notin PROD(k) \tag{3.23}$$

Where y^k is a binary variable which is equal to 1 if the interval k is selected and 0 otherwise.
In the general case, when the split factors $SM_f^{k,kk}$ are optimization variables, the bi-linear term $F_{OUT,f}^{i,k} \cdot SM_f^{k,kk}$ contained in eq. 3.22 makes this constraint non-convex (Bergamini *et al.*, 2005).

Single-stream flow model

The single-stream flow model represents a simplified case, which is obtained when the problem is subjected to the condition of selecting one process interval per process task. For this class of problems, a linear model formulation can be adopted, as described through equations 3.24-3.26.

$$F_f^{i,k,kk} \leq \sum_f (F_{OUT,f}^{i,k} \cdot \zeta_f^{k,kk}) \tag{3.24}$$

$$\sum_{kk} F_{i,k,kk} = \sum_f (F_{OUT,f}^{i,k}) \tag{3.25}$$

$$\sum_k \left(y^k \cdot \upsilon^{k,st}\right) \leq 1 \tag{3.26}$$

where st is the index of the process steps contained in the superstructure and $\upsilon^{k,st}$ is the data defining the allocation of intervals to steps (equal to 1 if interval k is allocated to step st and zero otherwise).

3.5 Economic models

In this section, some economic models used to calculate the financial part of the decision-making process are reported.

3.5.1 Operating Expenses

The operating expense $OPEX$ represent the expenses related to the operation of the process network, calculated as:

$$OPEX = R_{cost} + U_{cost} + W_{cost} + T_{cost} \tag{3.27}$$

Where R_{cost} is the cost for raw materials, U_{cost} for utilities and chemicals, W_{cost} the cost for disposal of wastes and T_{cost} the transportation cost, calculated as:

$$R^k_{cost} = \sum_{k \in RAW} \sum_i (F^{i,k}_{OUT} \cdot \pi^k_R) \tag{3.28}$$

$$U^k_{cost} = \sum_{f,i} (F^i_{ut,f} \cdot \pi^{ut,k}_U) \tag{3.29}$$

$$T_{cost} = \sum_{k,kk} (F^{i,k,kk} \cdot \pi^{k,kk}_T \cdot \eta^{k,kk}) \tag{3.30}$$

$$W^k_{cost} = \sum_{f,i} (F^i_{W,f} \cdot \pi^i_W) \tag{3.31}$$

where π^k_R is the price of raw material k, $\pi^{ut,k}_U$ is the price for utility ut in interval k, $\pi^{k,kk}_T$ is the price of transportation/ pumping between interval k and interval kk (in \$/km/kg for transportation or \$/bar/kg for pumping), $\eta^{k,kk}$ is the distance (expressed in km) or the pressure drop (in bar) between interval k and interval kk and π^i_W is the cost of disposal of component i.
The price associated to waste emission can either reflect the price for waste disposal, as for example in case of wastewater dischage to sewer system, or a penalty related to the waste emission, as for example in case of emission of greenhouse gases.

3.5.2 Gross revenues

The gross revenue $GREV$ is obtained through the sales of the products of the processing network, calculated as:

$$GREV = \sum_{k \in PROD(k)} (F^{i,k}_{OUT} \cdot \pi^{i,k}_P) \tag{3.32}$$

where $\pi^{i,k}_P$ is the price of product k with respect to its composition i, and $PROD(k)$ is the subset of k representing product alternatives, based on the allocation defined by $\nu^{k,st}$.

3.5.3 Capital investment

The capital investment represents the expense required for the construction of process intervals, calculated as:

$$CAPEX = \sum_{k}(INV^{k}) \tag{3.33}$$

where INV^{k} is the investment cost related to process interval k, which is calculated as a function of the throughput as:

$$INV^{k} = (\pi^{k}_{Ca} \cdot F_{k})^{\pi^{k}_{Cb}} \tag{3.34}$$

where π^{k}_{Ca} and π^{k}_{Cb} are coefficients for the capital investment model.

3.6 Objective functions

In investment projects and process optimization, commonly used objective functions are the maximization of financial indicators such as Gross Operating Income (GOI), Earning Before Interests and Tax (EBIT) and Net Present Value (NPV) (Perry and Green, 2008).
The GOI is a measure of the cash flow related to a certain operation, given as:

$$GOI = GREV - OPEX \tag{3.35}$$

where $GREV$ are the gross revenues and $OPEX$ the operating expenses.
The EBIT represents the yearly profit from a given operation, before calculating the interests on the capital investment depreciation, and taxes contribution. It is calculated as:

$$EBIT = GREV - OPEX - \frac{CAPEX}{\tau} \tag{3.36}$$

where $CAPEX$ is the capital investment and τ is the investment time horizon.
The NPV is a measure of the value that the investment brings to the company, and it can be defined as:

$$NPV = \sum_{t=1}^{\tau} \frac{GOI_{t} - CAPEX_{t}}{(1+dr)^{t}} \tag{3.37}$$

Where t is the time variable (in years), $CAPEX_{t}$ is the capital invested in year t, dr is the discount rate and τ is the investment time horizon, which can be tuned to reflect short-term or long-term optimization perspectives.
Depending on the design problem specifications, other objective functions can also be used, for example considering the simultaneous maximization of NPV and profitability, or including sustainability indicators such as carbon footprint, etc. (Douglas, 1988; Perry and Green, 2008).

4

Workflow for processing network synthesis and design

In this chapter, the workflow for problem formulation and solution is presented. This general structure of the workflow allows the formulation of:

- deterministic problems, for which no data uncertainty is considered
- stochastic problems, for which some of problem data are uncertain

The workflow for *stochastic problems* is constituted by 7 main steps. The first two steps are related to the definition and formulation of the design problem under uncertainty, which is then solved through steps 3-6. The last step deals with the analysis of the results, and their consolidation in a report, containing all results and information useful for the decision-making process, as well as problem documentation. The workflow for *deterministic problems*, on the other hand, requires the execution of steps 1 and 3 only, therefore steps 2 and 4-7 are skipped for these problems.
A schematic representation of the framework, in which the integration between workflow, dataflow, and supporting methods and tools is highlighted as reported in figure 4.1. In the next sections of this chapter, each step of the workflow is described. The supporting methods, tools and solution strategies are presented in chapter 5-7.

4.1 Step 1: Problem Formulation

In the first step of the workflow the synthesis and design problem is formulated, by defining its goal, objectives and scope, as well as by collecting and systematizing all necessary data and information. Being based on inputs with respect to all aspects of the problem (engineering, commercial, financial, etc.), different databases and expertise are required for its execution.

4.1.1 Problem definition

The first task of the workflow is the problem definition. The goal, objectives and scope of the problem are identified and stated. The objective function for the optimization problem is selected, based on the main goal of the synthesis and design problem.
All additional indicators which, together with the objective function, are relevant to assess the quality of the obtained solution are also identified at this point. These may include engineering indicators (e.g. raw material to product yield, specific energy consumption, etc.), sustainability indicators (e.g. carbon footprint, water footprint,

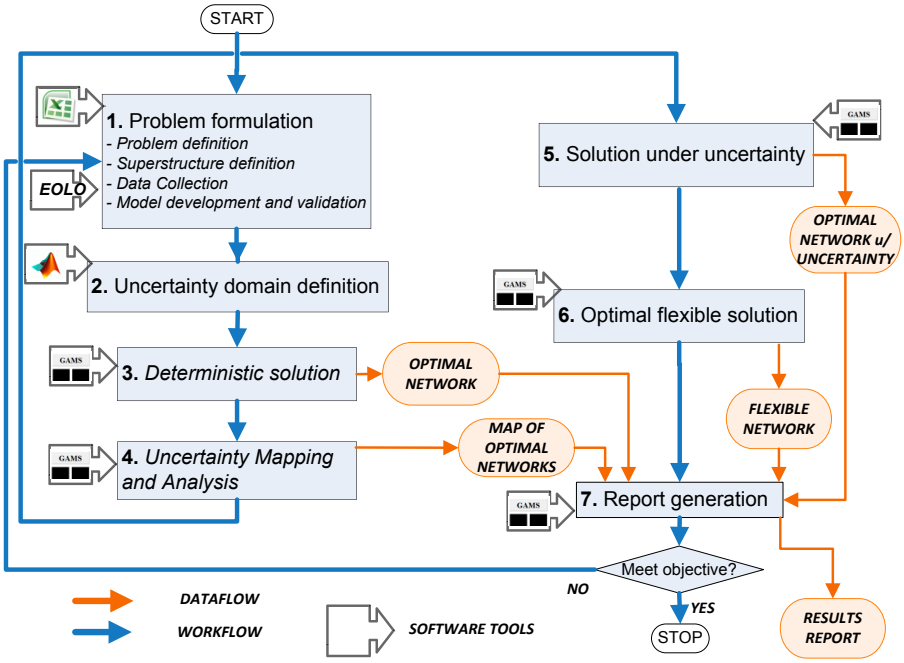


Figure 4.1. Schematic representation of the workflow: the integration between workflow, dataflow, and supporting methods and tools

etc.) or financial indicators (e.g. cash flow, return on investment, etc.).
All engineering, commercial or financial considerations, defining criteria for the acceptance or the success of the project are investigated, and converted to thresholds with respect to the objective function or indicators values, or constraints. Success criteria may be derived from financial constraints with respect of the investment performance (e.g. minimum acceptable return on investment, maximum acceptable capital investment, etc.), engineering constraints with respect to the complexity of the network (e.g. maximum number of processing steps or intervals, etc.) or commercial constraints with respect to the composition of the product portfolio (e.g. minimum acceptable sales of a given product, etc.).
Finally, the problem is identified as greenfield (if related to the design of a new processing network) or retrofit (in case of expansion or modification of an existing one).

4.1.2 Superstructure definition and data collection

Superstructure definition

After the problem is defined, the workflow proceeds with the specification of the alternatives existing with respect to raw materials, process and products, which constitute the search space for the design problem. As explained in section 2.1, this framework employs a superstructure representation of the alternatives, in which can-

didate process network structures are represented as a network of process intervals. For some problems (especially when new processes or products are concerned), the generation of the superstructure may result in a complex and time consuming task, and different methods can be used. In the interest of clarity, within this thesis those methods are classified as:

- alternative collection
- combinatorial synthesis
- insight-based synthesis

The *alternative collection* method consists of listing all known processing network configurations (based on previous experiences, known alternatives or scientific literature), and organize them in a superstructure representation. Examples of superstructure definition by alternative collection can be seen in the work of Rojas-Torres *et al.* (2013) and Khor *et al.* (2011).
The main drawback of this simple method is constituted by the possibility of including only known alternatives in the search space, hence not allowing to obtain innovative solutions. Furthermore, for the same reason this method does not allow to formulate problems related to new processes or products, for which production configurations are not known.
In *combinatorial synthesis* methods, the superstructure is generated by listing all relevant raw materials, products and process interval alternatives, and connecting them in all possible ways. This results in general in the definition of very large search space for the design problem, allowing the potential identification of innovative design. On the other hand, the combinatorial nature of the method may result in the generation of a large number of redundant configurations, in which the process elements contained in the superstructure are connected in implausible sequences (for example a product separation step prior to a product synthesis step). The enlargement of the search space results in an increase of computational effort required for the solution of the optimization problem, making this method arguably not suitable when large and complex problems are concerned. Examples of combinatorial superstructure generation can be found in the work of Tan *et al.* (2009) and Karuppiah and Grossmann (2006).
Finally, *insight-based synthesis* methods employ a synthesis algorithm, which use commercial and engineering insights as input for the systematic generation of the superstructure. The main feature of these methods is their ability to generate superstructures containing innovative solutions, while at the same time excluding those alternatives which, based on the available knowledge, are known to be unfeasible on non convenient.
By eliminating unfeasible and non convenient alternatives from the search space of the design problem, the computational resources needed for the solution of the optimization are reduced. Although more complex to implement from an algorithmic point of view, therefore, insight-based methods represent a convenient alternative for the formulation of large problems superstructure, especially when new products and processes are concerned. Although generally under-represented in the scientific literature, reference to these methods can be found in the work of Siirola (1996) and

Barnicki and Siirola (2004) .
In order to facilitate the task of superstructure generation, an insight-based synthesis method (based on means-ends analysis) has been developed and integrated in the framework. A complete description of the method will be given in section 6.1.

4.1.3 Data collection and systematization

Once the superstructure is defined, the workflow continues with the collection of data and information required for the formulation of the design problem. In order to facilitate the management of the large amount of data which are needed at this step, a data architecture has been developed and integrated within the developed framework. Such an architecture provides a structure for the systematization and storage of data, designed to facilitate the dataflow through the framework. The data architecture will be presented in section 6.2.
The list of raw materials, processing intervals and products is extracted from the superstructure, and the component list is defined, incorporating all process components and all components used to describe utilities and chemicals.
In most of the problems related to chemical synthesis, components are known and well identified, and consequently the component list definition results in a routine compilation task. In some cases, though, the design problem may be characterized by an extremely large number of non well-defined components. Typical examples of this type of problems include the design of processing network related to the treatment of natural products, such as in food processing, biotechnology and wastewater treatment. For these cases, the definition of a component list based on species identity is often impossible or impractical, due to the lack of data or to the complexity related to tracking extremely large number of species. Therefore, the component list is defined based on pseudo-components, in which classes of components are lumped based on properties or first-principles.
In order to simplify the task of component definition for wastewater network problems, this framework integrates a characterization method for wastewater composition, based on ASM components (Henze et al. 1987). The method will be described in chapter 6.3.
For each of the alternatives included in the superstructure, all relevant engineering, commercial and regulatory data and information are collected and systematized in a knowledge structure. Relevant data for raw materials and products include composition, quality specification, maximum and minimum flowrate, cost price, etc. Relevant data for processing options include yields, utility consumption, capital cost etc.
Because of the multidisciplinary nature of the design problem, in general different sources and databases (e.g. internal knowledge bases, market reports, scientific literature, etc) are consulted in order to collect the data. Therefore, particular emphasis has to be given to data reconciliation, in order to ensure the consistency of the formulated problem, and consequently the quality of the obtained solution. In order to facilitate this task, the above mentioned data architecture is designed in order to allow automatic execution of data consistency checks, which are used to identify inconsistency in the problem definition, which may result from faulty data specifications.

Finally, the knowledge and insights collected are used to analyze the superstructure, in order to detect eventual unfeasible or unwanted alternatives. If identified, those alternatives are eliminated from the search space through the definition of logical constraints, as explained in section 3.4.1.

4.1.4 Model selection, development and validation

The last task of the problem formulation step is the selection of models describing each of the elements of the superstructure, as well as of models required for the calculation of the objective function and of all indicators selected in step 1.
In order to facilitate this task, the integrated business and engineering framework is coupled with a model database. In particular, as described in chapter 3, the database contains a generic process interval model, through which a wide variety of problems can be formulated and solved in a generic manner. If needed, customized models can also be used within the framework structure.
The superstructure, the interval models, the objective function, the logical constraints and the variable bounds defined in the previous steps constitute an MILP or MINLP problem (equations 4.1 - 4.6).

$$\max_{x,y} f(x,y) \tag{4.1}$$

$$g(x,y) \geq 0 \tag{4.2}$$

$$h(x,y) = 0 \tag{4.3}$$

$$x \in X \tag{4.4}$$

$$x^{LO} \leq x \leq x^{UP} \tag{4.5}$$

$$y \in \{0;1\}^n \tag{4.6}$$

Where f is the objective function, x is the vector of continuous variables representing operating variables (e.g. flows) defined by their upper and lower bounds x^{UP} and x^{LO} in a continuous feasible region X, y is the vector of binary variables representing the selection of a process interval, g and h are the vectors of inequality and equality constraints, representing the process model and process specifications.

4.2 Step 2: Uncertainty domain definition

In step 2, the domain of uncertainty which is required as input for the optimization under uncertainty is defined. The execution of this step is required only for the formulation of stochastic problems, and it is therefore skipped when deterministic problems are concerned.
Within this step, uncertain data are identified and characterized in terms of statistical distributions. Monte Carlo sampling of these distributions is then performed, resulting in the definition of a number of future scenarios, each representing a possible realization of the uncertain data.

4.2.1 Selection of uncertain data

While in principle almost all data are associated to a certain degree of uncertainty, in general not all of the sources of uncertainty will have significant impact on the design problem. Moreover, when large problems are considered, it is not possible or convenient to consider them all in the optimization under uncertainty problem, because of the associated mathematical complexity. Therefore, the first step in the definition of the uncertainty domain is constituted by the selection of the data, which will be considered uncertain in the optimization problem.
Different considerations can be used as premises for such a selection. Data may be classified as uncertain because of their intrinsic uncertain nature (such as forecast of prices, market sizes, etc.), or based on evidences obtained in the data collection step (for example when different sources report different values for the same data). In other cases, those data are obtained from off line solution of models (e.g. the estimation of physical properties from thermodynamic models), to which a certain degree of uncertainty is associated. Alternatively, heuristics and experience may drive the selection, for example in order to include data which have shown significant uncertainty in similar cases. Finally, the selection may be done in order to study a specific aspect of the problem, which the user wants to investigate.

4.2.2 Uncertainty specification

The uncertainty associated with each of the selected data is described in terms of probability distribution. For each data, a probability density function representing the uncertain distribution (e.g. normal, uniform, etc.) is selected, and its specific parameters defined (e.g. mean and variance for normal distribution; maximum and minimum bounds for uniform distribution). Additionally, the correlation between the uncertain data is analyzed, and characterized in terms of covariance between the probability distributions of each couple of data.
When possible, this characterization is performed by statistical analysis of experimental data or historical observations. Unfortunately, for some of the data commonly used in this class of enterprise-wide optimization problems (e.g. prices and market sizes of new products, performances of new processes etc.), such observations are not available. In these cases, an expert review approach is usually followed (Helton and Davis, 2003; Sin *et al.*, 2009), in which the uncertain data are assigned to different classes of uncertainty (e.g., low, medium and high) where each class has a predefined distribution with its variance and covariance.

4.2.3 Monte Carlo sampling

The domain of uncertainty defined in this step is sampled to generate a list of possible future scenarios, with equal probability of realization.
In the further step of the workflow, the generated Monte Carlo samples are used as discretization points to approximate the probability integral, appearing in the objective function of optimization under uncertainty problems. Consequently, particular emphasis is given to the generation of samples which assure a uniform and

representative coverage of the uncertain space, in order to limit the error introduced by the approximation (Birge and Louveaux, 1999).
In order to facilitate this task, and assure the quality of the sampling procedure (in terms of coverage of the uncertain space) the framework integrates a Latin Hypercube Sampling (LHS) based sampling technique. The rank correlation control method proposed by Iman and Conover (1982) is employed, in order to reflect the correlation between the uncertain parameters in the generated future scenarios.

4.3 Step 3: Deterministic formulation and solution

In this step, the problem is solved in deterministic conditions.

- for deterministic problems: this corresponds to the solution of the problem defined in step 1 of the workflow, represented by equations 4.1 - 4.6. As a result, the solution of the synthesis and design problem is obtained, by identifying the optimal selection of raw materials, product portfolio and process configuration, together with the material flow through the network as well as the value of all variables free for optimization. The workflow for deterministic problems is therefore concluded.

- for stochastic problems: this corresponds to the solution of the optimization problem for the expected realization of the uncertainty. The problem formulation is obtained by disregarding data uncertainty and fixing the uncertain data to their expected value.

4.4 Step 4: Uncertainty mapping

In step 4, the consequences of the data uncertainty on the decision-making problem are investigated and mapped. To this goal, a separate deterministic optimization problem is solved for each of the scenarios generated by Monte Carlo sampling in step 2. According to the stochastic programming terminology, this corresponds to the solution of a family of *wait-and-see* optimization problems (Birge and Louveaux, 1999). The main characteristic of *wait-and-see* optimization is the assumption that the decision-maker has the possibility to wait until the realization of the uncertain data is observable, and use this observation in order to take the decision (in opposition to *here-and-now* problems, in which the decisions have to be taken prior to the observation).
When the design of a processing network is concerned, the *wait-and-see* assumption is in general non valid, since decision-makers have to take design decisions prior to be able to observe future data. Nevertheless, some insights into potential consequences of uncertainties and relevant indications can be obtained through the execution of this step.
The result of the uncertainty mapping step is constituted by a distribution of optimal processing networks and of objective function values, obtained as separate

wait-and-see solutions for different realization of the uncertain data. While these distributions do not constitute the solution of the overall design problem, relevant information can be obtained through their analysis.
The number of different processing network topologies which are obtained as solutions for the different scenarios, for example, gives an indication of the impact of the data uncertainty specified in step 2 on the decision-making problem. Moreover, the frequency associated to the selection of each individual processing network is observed, and can be used as rationale for the simplification of problems whose size or complexity makes it impossible to solve in the complete form. An example of simplification strategy based on uncertainty mapping result is constituted by the superstructure reduction policy, which will be presented in section 5.6. Finally, the expected value of the distribution of objective function values represents an upper bound for the value of the solution under uncertainty (in case of maximization), which can be used to identify and stop projects whose performances are below the requirements, prior to the execution of the rest of the analysis.
Although based on a simplification of the design problem under uncertainty, the above mentioned analysis allows estimating the role of the considered data uncertainty in the decision-making problem, without requiring a large computational investment, and constitute therefore a useful step in problem workflow.

4.5 Step 5: Solution under uncertainty

In this step, the domain of uncertainty defined in step 2 is incorporated in the design phase, through the formulation and solution of the problem of optimization under uncertainty.
The problem is formulated as a two stage stochastic programming problem. Through this formulation, the design problem is considered as resulting of two stages. In the first stage, the exact realization of the uncertain data is unknown, and the design decisions (termed first stage decisions in stochastic programming terminology) are taken, based on the knowledge of the probability distribution of the uncertain data. In the second stage the uncertainty is disclosed and the exact value of the uncertain data becomes known. On the base of this additional knowledge, corrective actions (termed second stage decisions) are taken. In a design problem such as the one formulated here, the set of future scenarios generated by Monte Carlo sampling in step 2 is used in order to simulate this second stage. When processing networks are concerned, corrective actions consist typically in modifications of the operating conditions (within the flexibility limits imposed by design decisions taken) in order to adapt the network to the realization of the uncertain scenario, hence obtaining feasible and optimal operations. Consequently, through the solution of this problem, a processing network design which is feasible over the entire uncertain space and whose expected objective function value is optimal, is identified.
From a mathematical standpoint, the formulation of two stage problems requires classifying variables in first and second stage, with respect to the stage at which their value has to be fixed. For this class of problems, first stage variables typically correspond to design decisions, such as the topology of the network and the

design values for material flows and process conditions; second stage variables are constituted by operational decision, such as actual material flows and processing conditions.
In equations 4.7 - 4.14, the two stage stochastic programming problem is reported, in its deterministic equivalent formulation (Birge and Louveaux, 1999).

$$\max_{x,y} f_I(x_I, y) + E_\Theta \left(f_{II}(x_I, x_{II}, y, \Theta)\right) \tag{4.7}$$

$$s.t. g(x_{II}, y, \Theta) \geq 0 \tag{4.8}$$

$$h(x_{II}, y, \Theta) = 0 \tag{4.9}$$

$$p(x_I, x_{II}, y, \Theta) \geq 0 \tag{4.10}$$

$$q(x_I, x_{II}, y, \Theta) = 0 \tag{4.11}$$

$$x^{LO} \leq x \leq x^{UP} \tag{4.12}$$

$$y \in \{0; 1\}^n \tag{4.13}$$

$$\Theta \in [\Theta^{LO}, \Theta^{UP}] \tag{4.14}$$

where f_I and f_{II} represents the objective function components with respect to first and second stage decisions, Θ is the vector of uncertain data and $E_\Theta(f)$ is the expected value of the objective function f over the Θ space.
Equations 4.10-4.11 enforce limits to the values of second stage variables, based on the design decisions taken as first stage variables, hence representing the limits for the corrective actions represented by the design decision. For example, the flow through a process or equipment (second stage variable) has to be less or equal than the maximum design flow for that process or equipment (first stage variable). Through these constraints, the link between the two stages of decision making process is established.
The calculation of the expected value of the objective function (equation 4.7) requires the evaluation of a multidimensional probability integral. For large problems considering many uncertain parameters, the evaluation of this integral may result in a cumbersome and complex procedure, and require a large computational investment. A common approach in stochastic programming is based on a Monte Carlo sampling based method for the approximation of the expected value of the objective function (equation 4.15). This strategy is generally known as Sample Average Approximation (SAA) (Birge and Louveaux, 1999).

$$E_\Theta[f(x, y, \Theta)] \approx \sum_{s=1}^{NS} (P_s \cdot f(x, y, \Theta_s)) \tag{4.15}$$

$$s.t. g_s(x, y, \Theta_s) \geq 0 \;\; \forall s \in S \tag{4.16}$$

$$h_s(x, y, \Theta_s) = 0 \;\; \forall s \in S \tag{4.17}$$

where s is the index of Monte Carlo samples (representing possible future scenarios), NS is the number of samples, Θ_s is the realizations of Θ in sample s and P_s the probability of realization of sample s.
It is important to underline that, when the problem is reformulated to the SAA,

each constraint containing an uncertain data is converted into NS constraints, each one of them related to a different sample s (equations 4.16 - 4.17). Consequently, the number of constraints constituting the MINLP problem is increased, by a maximum factor NS. This results in an increase in problem complexity, and consequently in the computational resources needed for its solution.
Within the framework, this additional complexity is managed through variable initialization and bounding, which is performed based on the solutions obtained in the previous steps of the workflow. Moreover, the framework integrates a bi-level decomposition scheme, which can be adopted for the solution of large complex problems, involving large number of uncertain data. A more detailed explanation of the solution algorithm based on the bi-level decomposition is reported in section 5.2.2.
The results identify the network which is feasible over the entire uncertain space and whose expected objective function value is optimal. This solution is termed *optimal network under uncertainty*.
The use of expected value as conditional value for objective function calculation corresponds to a risk-neutral approach to decision-making (Shapiro, 2012). Other conditional values (for example, *value-at-risk*) can be used instead of the expected value, to reflect different attitudes with respect to risk.

4.6 Step 6: Optimal flexible solution

As explained in the previous section, because of the additional knowledge available at operational stage (when some of the uncertain data become observable) the operational policy defined in the design stage may become sub-optimal, and a new optimal operational policy may exist. The problem of identifying the optimal operational policy with respect to production volumes, patterns etc. for an existing facility or network (i.e. within the flexibility allowed by a given design) is defined as planning problem (Erdirik-Dogan *et al.*, 2007; Ierapetritou *et al.*, 1996b; Terrazas-Moreno and Grossmann, 2011).
It is evident that, the higher the flexibility allowed by the design of a network, the higher the possibility to modify the operational policy and to adapt to a different realization of the uncertain data will be, once these become observable. At the same time, higher design flexibility is often obtained at the expense of a larger capital investment (for example through redundant design or over-design). As a consequence, a trade-off exists between the capital investment and the ability of mitigating negative consequences of the uncertainty at planning and operational stage. In the industrial design practice, the level of design flexibility is determined on the basis of previous experience, heuristic or safety factors.
In this step, a different formulation of the two stage stochastic programming is solved, aiming at the identification of the optimal trade-off between operational flexibility and capital investment. To this end, the binary variables associated to the selection of process intervals (which were considered as first stage variables in the previous step) are also partitioned in first and second stage variables. Consequently, the search space for the design problem is enlarged, including *redundant* processing network configurations, meaning configurations having the flexibility to assume different operational topologies (with respect to selection of raw materials,

processing path and product portfolio), as a result of second stage decisions. Within this framework, such a solution is termed *optimal flexible network*.
The identification of the optimal flexible network requires the solution of a two stage stochastic programming, in which binary variables are included as second stage decisions. The deterministic equivalent formulation of the design problem is reported in equations 4.18-4.25 (Birge and Louveaux, 1999).

$$\max_{x,y} f_I(x_I, y_I) + E_\Theta \left(f_{II}(x_I, x_{II}, y_I, y_{II}, \Theta) \right) \tag{4.18}$$

$$s.t. g(x_{II}, y_{II}, \Theta) \geq 0 \tag{4.19}$$

$$h(x_{II}, y_{II}, \Theta) = 0 \tag{4.20}$$

$$p(x_I, y_I, x_{II}, y_{II}, \Theta) \geq 0 \tag{4.21}$$

$$q(x_I, y_I, x_{II}, y_{II}, \Theta) = 0 \tag{4.22}$$

$$x^{LO} \leq x \leq x^{UP} \tag{4.23}$$

$$y \in \{0; 1\}^n \tag{4.24}$$

$$\Theta \in [\Theta^{LO}, \Theta^{UP}] \tag{4.25}$$

where the subscript I indicates first stage variables and II second stage variables.
With respect to the problem formulated in the previous step, the possibility of adopting different operational topologies as result of second stage decisions results in an increase of the number of binary variables by a factor NS. Moreover, additional constraints are added in order to link the two stages of decision making. For example, the selection of a process interval as part of the topology defined at the operational stage (second stage decision) is of course subject to its selection as part of the processing network defined at design stage (first stage decision). Because of the increase in search space and in number of variables and constraints, the mathematical solution of the problem associated to the design of the optimal flexible network may be challenging and require large amount of computational time and resources.
In order to facilitate the execution of this step and cope with the increased complexity, the framework employs previous solutions in order to provide good variable initialization and bounding. Moreover, the above mentioned bi-level decomposition strategy (see section 5.2.2), and the superstructure reduction policy (see section 5.3) can be used to solve the problem.

4.7 Step 7: Report generation

In the last step, the results obtained in the previous steps of the design framework are analyzed and aggregated in a results report, containing a documentation section, a results section and a summary section.

4.7.1 Documentation section

The documentation section contains all information needed to reproduce the design problem (superstructure, models, data and uncertainty domain definition), providing a complete documentation of the results obtained, as well as a structure in which all the collected information can be systematized and stored for future reference and use.

4.7.2 Results section

The results section is constituted by a collection of all the results obtained through steps 3 to 6. The solutions obtained for each step are included in this section, by reporting all variable optimal values in the form of data tables.

4.7.3 Summary section

The summary section contains an overview of the obtained results (in terms of network structure and objective function value), as well as the value of indicators calculated from those results, constituting an aggregate input to the decision-makers. It is important to underline that all results and indicators are obtained under a user-specified domain of uncertainty (defined in step 2). When those results are used to take real decisions, therefore, the domain of uncertainty under which those have been obtained must be considered, since it defines the range of validity for the analysis. In particular, the summary section include indicators related to the the consequences of uncertainty on the decision making process such as (Birge and Louveaux, 1999):

- Expected Value of Perfect Information (EVPI)
- Value of Stochastic Solution (VSS)
- Uncertainty Penalty (UP)

Expected Value of Perfect Information (EVPI)

The EVPI is given as:

$$EVPI = E_\theta(\max_{x,y}(f(x,y,\theta))) - \max_{x,y}(E_\theta(f(x,y,\theta))) \tag{4.26}$$

Where, as explained in the previous sections, the first term is the expected value of the distribution of objective functions obtained in the uncertainty mapping stage (step 4 result), and the second term is the objective function value obtained for the optimal network under uncertainty (step 5). The EVPI quantifies the cost of not knowing the exact value of the uncertain data, calculated as the expected reduction in objective function value which is caused by the uncertainty. The quantification of the cost associated to the lack of exact knowledge given by the EVPI constitutes an extremely relevant information. As described in the introduction, processing network design projects are in general developed through a stage-gate process, called development funnel. Through the funnel, the project is developed in a succession of stages, the design is progressively refined, and the uncertainty is reduced by

conducting more detailed investigations of the problem (via customer assessments, lab and pilot scale experiments, rigorous modeling studies, etc.).
The decision of graduating a project from one phase to the successive one is in general taken based on the judgement of the project manager, in the attempt of finding a trade-off between the level of detail and precision of the analysis, and the time and resources limitations which are typical of engineering projects (Wheelwright, 1992). By providing a quantification of the value loss due to lack of information, the EVPI provides a value-based estimation of the maturity of a given project, which can be used to support the decision of transferring a project to the next stage of the development funnel. A large value of EVPI (relative to objective function value) suggests to keep the project in the exploratory phase (also called Front End Loading in process management terminology), and to perform further investigations, aiming (if possible) at reducing the uncertainty associated with the data, since this promises to largely benefit the performances of the network. Similarly, low EVPI values indicate more mature projects, which could be advanced to further stages of development. Finally, the EVPI gives an indication of the upper bound of the cost which such an exploratory phase should have.

Value of Stochastic Solution (VSS)

The VSS is given as:

$$VSS = \max_{x,y}(E_\theta(f(x,y,\theta))) - E_\theta((f(x^*_{det}, y^*_{det}, \theta))) \tag{4.27}$$

where the first term is the solution of the problem under uncertainty (step 5 and 6), and the second term is calculated by evaluating the performances of the network selected under deterministic conditions against the uncertainty conditions. In order to calculate this second term, a NLP or LP problem under uncertainty (obtained by fixing the value of first stage decisions to the results of step 3 in the problem defined by equations (4.7 - 4.14) is solved at this step.
The VSS quantifies the difference in performances between the implementation of the stochastic and the deterministic solutions, both evaluated under the domain of uncertainty defined in step 2. Such an indicator evaluate the value associated to performing the optimization under uncertainty, hence providing an estimation of the payback obtained through the consideration of the uncertainty in the decision making process.

Uncertainty Price (UP)

The UP is give as:

$$UP = \max_{x,y}(f(x,y)) - \max_{x,y}(E_\theta(f(x,y,\theta))) \tag{4.28}$$

where the first term is the solution of the deterministic problem (step 3), and the second term is the solution under uncertainty (step 5 and 6). The UP quantifies the reduction in performances which is associated to the necessity of considering uncertainty in the data, and therefore provides an estimation of the cost of the uncertainty.

5

Solution Strategies

In this chapter, the solution strategies integrated in the framework are described. A general overview of these solution strategies, organized with respect to the class of problems they are suitable to solve is shown in table 5.1.

Table 5.1. Overview of the solution strategies for the different classes of problems

	Single-stream	**Multi-stream**
Deterministic	Direct (5.1) Bi-level decomp. (5.2.1)	Direct (5.1) Bi-level decomp. (5.2.3)
Stochastic	Direct (5.1) Bi-level decomp. (5.2.2) Superstr. reduction (5.3)	Bi-level decomp.* Superstr. reduction*

*whithin this Ph.D. project, the solution of multi-stream stochastic problems has not been obtained.

5.1 Direct Solution

The direct solution is the simplest solution method, and it is employed whenever the size and complexity of the problem allows it. Depending on the type of problem to be solved, different solvers are employed:

- Linear problems are solved through the Simplex algorithm implemented in the solver CPLEX version 12 (IBM Corp., 2009).
- Convex non-linear problems are solved via Outer Approximation /Equality Relaxation (OA/ER), which is implemented in the solver DICOPT (Viswanathan and Grossmann, 1990).
- Non-convex problems are solved through the Branch-and-Reduce algorithm implemented in the solver BARON (Tawarmalani and Sahinidis, 2005).

The above mentioned solvers are implemented in GAMS, and constitute the state of the art of solution methods for discrete optimization problems. For a detail description of these algorithms, the reader is invited to refer to the above mentioned references.

5.2 Bi-level decomposition method

Bi-level decomposition is a commonly used solution scheme for complex optimization problems, especially when the problem structure is characterized by the existence of a certain number of constraints (e.g. non-linear, non-convex) that complicates the solution (Erdirik-Dogan *et al.*, 2007; Kopanos *et al.*, 2009; Terrazas-Moreno and Grossmann, 2011). Those constraints are often referred to as *complicating constraints*.
This solution strategy is based on the decomposition of the problem into a relaxed and a local sub-problem. The relaxed sub-problem is obtained through the relaxation of complicating constraints, resulting in a simpler problem formulation, which is solved to obtain an upper bound (in case of maximization) for the objective function value of the original problem.
The local sub-problem is obtained by fixing the value of some decision variables (typically the binary variables) to the solution of the relaxed sub-problem, to obtain a local solution of the original problem, which represents a lower bound (in case of maximization) for the objective function of the original problem.
In bi-level decomposition schemes, the 2 sub-problems are solved iteratively, adding cuts at each iteration to exclude previously obtained solutions from the search space. The algorithm continues producing a succession of upper and lower bounds for the solution, until a stopping criterion is met. Typical stopping criteria are related to the convergence of the two bounds or to the number of iterations performed.
The effectiveness of this solution approach is highly dependent on the tightness of the relaxation used to define the upper level problem. In the worst case from a computational perspective, bi-level decomposition approaches may require the complete exploration of the solution space, before the global optimum is identified (Erdirik-Dogan *et al.*, 2007).
A schematic representation of a generic solution algorithm based on bi-level decomposition is reported in figure 5.1.
The developed business and engineering framework integrates bi-level decomposition schemes, which can be used for the solution of optimization problems formulated through the different steps of the workflow, and whose direct solution is not possible or inconvenient, due to the size and complexity of the problem. In particular, the framework integrates bi-level algorithms designed for the solution of 3 classes of problems:

- problems with non-linear capital cost constraints
- two stage stochastic problems
- multi-stream problems

In the next sections, these solution methods are described.

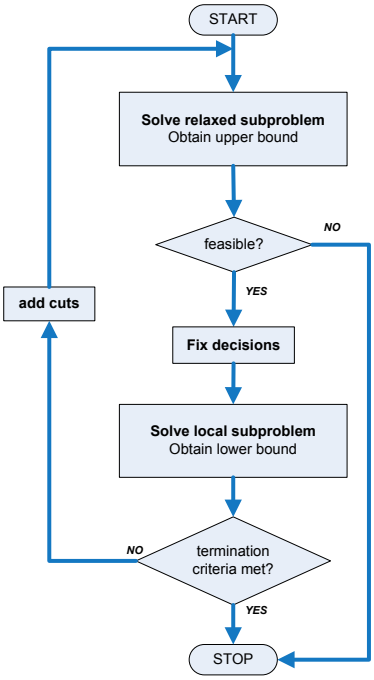


Figure 5.1. General bi-level decomposition algorithm (for maximization problems)

5.2.1 Bi-level decomposition for non-linear capital cost constraints

As described in section 3.5, this framework integrates a capital cost model, which estimates the capital cost associated to the construction of each process interval as a power function of the throughput (equation 3.34). For single-stream problems based on the generic process interval model presented in chapter 3, the capital cost model represents the only non-linear constraint in the MINLP formulation.
In order to facilitate the solution of such a problem, a bi-level decomposition scheme is developed.

Relaxed sub-problem

The relaxed sub-problem is obtained through the relaxation of the capital cost constraint (equation 3.34), which is substituted by piecewise linear underestimators, as described by equation 5.1 - 5.3.

$$INV^k = \sum_{j=1}^{N_L} \left(\widehat{\pi}_{Ca}^{k,j} \cdot F_D^{k,j} + \widehat{\pi}_{Cb}^{k,j} \cdot v^{k,j} \right) \tag{5.1}$$

$$\Gamma^{k,j} \cdot v^{k,j} \leq F_D^{k,j} \leq \Gamma^{k,j+1} \cdot v^{k,j} \tag{5.2}$$

$$\sum_{j=1}^{N_L} \left(v^{k,j}\right) = 1 \tag{5.3}$$

where INV^k is the capital cost for interval k, j is the index indicating the linearization intervals, N_L is the number of intervals, $v^{k,j}$ is the binary variable indicating the interval containing the flow; $F_D^{k,j}$ is the variable representing the disaggregated flow F^k over the grid $\Gamma^{k,j}$; $\widehat{\pi}_{Ca}^{k,j}$ and $\widehat{\pi}_{Cb}^{k,j}$ are the coefficients for the piecewise linearization of the capital cost constraint, which are calculated prior to the optimization as:

$$\widehat{\pi}_{Ca}^{k,j} = \left((\pi_{Ca}^k \cdot \Gamma^{k,j+1})^{\pi_{Cb}^k} - (\pi_{Ca}^k \cdot \Gamma^{k,j})^{\pi_{Cb}^k}\right) \cdot \left(\Gamma^{k,j+1} - \Gamma^{k,j}\right)^{-1} \tag{5.4}$$

$$\widehat{\pi}_{Cb}^{k,j} = (\pi_{Ca}^k \cdot \Gamma^{k,j})^{\pi_{Cb}^k} - \widehat{\pi}_{Ca}^{k,j} \cdot \Gamma^{k,j} \tag{5.5}$$

The piecewise linear underestimator of the capital cost constraint is shown in figure 5.2.

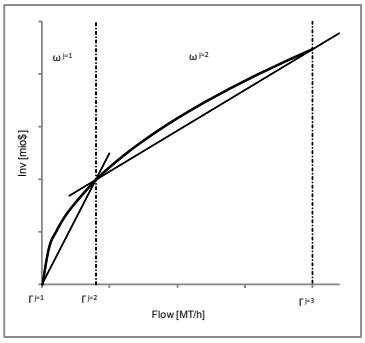


Figure 5.2. Schematic representation of the piecewise linear underestimator of capital cost over 2 intervals

Local sub-problem

The local sub-problem is obtained from the original problem by fixing the topology (binary variables) to the results of the relaxed problem.

Algorithm

As shown in figure 5.1, the two sub-problems are solved iteratively, adding cuts at each iteration in order to exclude previous solutions from the search space. At each iteration, the local sub-problem is solved for a fix topology, using the solution of the relaxed sub-problem as initial point. The procedure stops when a termination criterion is met. The best local solution is reported as the result, together with the associated optimality gap.

Note 1 *In most of the cases studied within the scope of this project, it has been observed that when 5 or more linearization intervals are considered, the piecewise underestimator is extremely tight, and the decomposition has shown the ability to converge within the first iteration, with extremely small optimality gap.*
Although this behavior cannot be generalized, it provides an indication of the tightness of the relaxation which can be achieved with piecewise linearization of this kind of constraints.
Consequently, piecewise linearization of capital cost constraints can in some cases represent an alternative to the original non-linear cost model, and allow simplifying the formulation of the problem, at the expense of a moderate or negligible error.

5.2.2 Bi-level decomposition for two stage stochastic problems

The second solution algorithm based on bi-level decomposition is dedicated to the solution of the two stage stochastic programming problems, formulated in step 5 and 6 of the workflow (section 4.6).
As described in the previous chapter, this class of problems is characterized by the existence of constraints linking the two stages of the decision making process, such as equations 4.16 - 4.17. As the incidence matrix reported on the left side of figure 5.3 shows, being associated to a large number of variables, those constraints complicate the solution procedure.

Relaxed sub-problem

As the structure of the problem suggests, the relaxed sub-problem is obtained through the relaxation of the complicating constraints (equations 4.16 - 4.17).
As the incidence matrix (figure 5.3) indicates, the structure of the relaxed sub-problem allows its further decomposition into a series of smaller sub-problems, which can be easily solved in a specific sequence (Ierapetritou *et al.*, 1996a). It should be noted that, if this second decomposition is performed, the resulting sub-problems are equivalent to the one solved as uncertainty mapping (step 4 of the framework).

Local sub-problem

The local sub-problem is obtained from the original problem by fixing the first stage variables (as defined in the previous chapter) to the results of the relaxed sub-problem.

Algorithm

Similarly to the previous case, the two sub-problems are solved iteratively, adding cuts at each iteration in order to exclude previous solutions from the search space. At each iteration, the solution of the relaxed sub-problem is used as initial point

for the local sub-problem. The procedure stops when a termination criterion is met. The best local solution is reported as the result, together with the associated optimality gap.

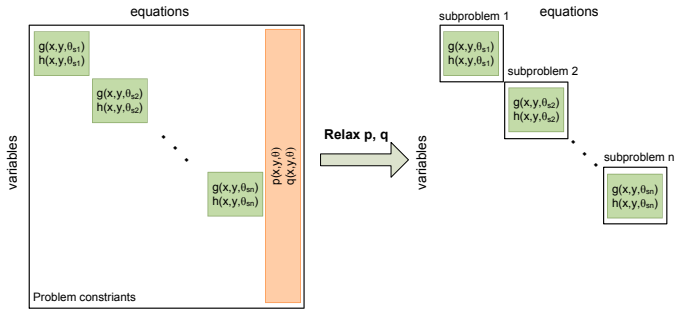


Figure 5.3. Bi-level decomposition for two stages solution

5.2.3 Bi-level decomposition for multi-stream problems

The third bi-level decomposition scheme is designed for the solution of large multi-stream problems, which cannot be solved directly with Baron. As described in chapter 3, multi-stream problems require the formulation of non-convex equations 3.22, which are complicating constraints.

Upper Bound

The Upper Bound problem is obtained through McCormick relaxation of the bi-linear constraints (McCormick, 1976). In this approach, the bi-linear constraints (equation 3.22) are substituted by the linear over- and underestimators:

$$F_f^{i,k,kk} \geq F_{out,f}^{i,k^{LO}} \cdot SM_f^{k,kk} + F_{out,f}^{i,k} \cdot SM_f^{k,kk^{LO}} - F_{out,f}^{i,k^{LO}} \cdot SM_f^{k,kk^{LO}} \tag{5.6}$$

$$F_f^{i,k,kk} \geq F_{out,f}^{i,k^{UP}} \cdot SM_f^{k,kk} + F_{out,f}^{i,k} \cdot SM_f^{k,kk^{UP}} - F_{out,f}^{i,k^{UP}} \cdot SM_f^{k,kk^{UP}} \tag{5.7}$$

$$F_f^{i,k,kk} \leq F_{out,f}^{i,k^{UP}} \cdot SM_f^{k,kk} + F_{out,f}^{i,k} \cdot SM_f^{k,kk^{LO}} - F_{out,f}^{i,k^{UP}} \cdot SM_f^{k,kk^{LO}} \tag{5.8}$$

$$F_f^{i,k,kk} \leq F_{out,f}^{i,k^{LO}} \cdot SM_f^{k,kk} + F_{out,f}^{i,k} \cdot SM_f^{k,kk^{UP}} - F_{out,f}^{i,k^{LO}} \cdot SM_f^{k,kk^{UP}} \tag{5.9}$$

where $F_{out,f}^{i,k,kk^{LO}}$ and $F_{out,f}^{i,k,kk^{UP}}$ are the lower and upper bound for $F_{out,f}^{i,k,kk}$, while $SM_f^{k,kk^{LO}}$ and $SM_f^{k,kk^{UP}}$ are the lower and upper bounds for the variable $SM_f^{k,kk}$. The result is a convex MINLP (if the problem formulation contains other non-linear constraints) or MILP formulation of the upper bound problem.

Local sub-problem

The local sub-problem is obtained from the original problem by fixing the binary variables to the topology obtained as result of the relaxed sub-problem.

Algorithm

The same solution algorithm described for the capital cost case is employed.

Note 2 *As many authors in the scientific literature report, McCormick-based relaxation of bi-linear terms are in general non-tight, resulting in slow convergence of the solution algorithm and in large optimization gaps (Bergamini* et al.*, 2005; Karuppiah and Grossmann, 2006; Bogataj and Kravanja, 2012).*

In order to tackle the issue reported in note 2, a number of strategies designed to tighten the McCormick relaxation and therefore to improve the performances of the solution method have been integrated in the computer-aided framework, such as:

- variables bounding
- strengthening cuts
- domain partitioning
- multi-level decomposition

Variables bounding

The variables bounding strategy consists of establishing tight upper and lower bounds for the continuous variables appearing in the bi-linear term.
When large problems are considered, the determination of the most appropriate value for those bounds may result in a complicated task. In order to facilitate the solution of the problem without affecting the quality of the solution, those bounds should be fixed as tight as possible (in order to improve the quality of the McCormick relaxation). At the same time, those bounds should not be too tight, in order to avoid eliminating feasible solutions from the search space.
In the proposed problem formulation, the split variables $SM_f^{k,kk}$ are by definition bounded between 0 and 1. The flow variables $F_{out,f}^{i,kk}$, on the other hand, are defined as positive, but do not have a clearly defined upper bound.
In order to tackle this issue, the framework integrates a problem data analysis tool, designed to identify good bounds for the flow variables in a systematic manner. The tool can be employed for all superstructure sections not containing internal or external recycles.
It can easily be seen that the maximum value of $F_{out,f}^{i,kk}$ is obtained when all flows containing the component i are fed to the interval kk. As a consequence, when the generic process interval model formulation is employed, the upper bound for the

flow variable $F_{out1}^{i,k^{UP}}$ can be calculated according to equation 5.10 for intervals representing material sources (e.g. raw materials), and via equation 5.11 for treatment intervals.

$$F_{out1}^{i,k^{UP}} = \phi^{i,k} \quad \forall k \in RAW(k) \tag{5.10}$$

where $\phi^{i,k}$ is the raw material flow of component i for interval k.

$$F_{out1}^{i,kk^{UP}} = \sum_{k} (F_{out1}^{i,k^{UP}} \cdot \zeta_{P}^{k,kk} + F_{out2}^{i,kUP} \cdot \zeta_{S}^{k,kk}) \cdot [(1 + \alpha^{i,kk} \cdot \mu^{i,kk}) \cdot (1 + \sum_{rr,react} (\gamma^{i,kk,rr} \cdot \theta^{react,kk,rr})) \cdot (1 - \delta^{i,kk}) \cdot \sigma^{i,kk}] \quad \forall kk \notin RAW(kk) \tag{5.11}$$

where $\zeta_{f}^{k,kk}$, $\alpha^{i,kk}$, $\mu^{i,kk}$, $\gamma^{i,kk,rr}$, $\delta^{i,kk}$, $\theta^{react,kk,rr}$ and $\sigma^{i,kk}$ are the data defining the connections in the superstructure, utility consumption, reaction, waste separation and separation respectively.
Similarly, for $F_{out2}^{i,k^{UP}}$:

$$F_{out2}^{i,k^{UP}} = \phi^{i,k} \quad \forall k \in RAW(k) \tag{5.12}$$

$$F_{out2}^{i,kk^{UP}} = \sum_{k} (F_{out1}^{i,k^{UP}} \cdot \zeta_{P}^{k,kk} + F_{out2}^{i,kUP} \cdot \zeta_{S}^{k,kk}) \cdot [(1 + \alpha^{i,kk} \cdot \mu^{i,kk}) \cdot (1 + \sum_{rr,react} (\gamma^{i,kk,rr} \cdot \theta^{react,kk,rr})) \cdot (1 - \delta^{i,kk}) \cdot (1 - \sigma^{i,kk})] \quad \forall kk \notin RAW(kk) \tag{5.13}$$

It should be underlined that the proposed analysis is based on simple function evaluations based exclusively on problem data, and can therefore be executed prior to the optimization, without any additional computational burden for the optimizer.
Moreover, in case the above mentioned analysis returns an upper bound equal to zero for a flow variable, the variable can be fixed to zero and the corresponding bi-linear constraint can be eliminated from the model, resulting in a reduction of the mathematical complexity of the problem. Being based on a systematic analysis of the problem data, this simplification is an exact method, and do not cause any consequence on the quality of the obtained solution.
As previously mentioned, the described tool can be employed only for a portion of the superstructure not included in recycles. When recycle streams exist, the tool can be applied in order to calculate the upper bound based on a user defined maximum recycle ratio. As an alternative, engineering insights can be used in order to determine the upper bounds (e.g. based on maximum flow through a process interval, solubility, etc.).

Strengthening cuts

Strengthening cuts are linear constraints, which were redundant in the original MINLP formulation. Due to the relaxation of the non-convex constraints, those cuts become non-redundant for the lower bound sub-problem, and may therefore

be included in order to tighten the relaxation and reduce the computational time (Karuppiah and Grossmann, 2006).
Examples of strengthening cuts are constituted by splitter mass balances, such as equation 3.25. Other strengthening cuts can be derived from the logical conditions existing with respect to process interval selection. In particular, in order to assure material flow through the network, the selection of a certain process interval (not representing a raw material or a product) implies that at least one of the intervals upstream and one of the intervals downstream connected to it must be selected.
The logic with respect to the upstream selection can be formulated as:

$$y^{kk} \leq \sum_{k} \left(\zeta_P^{k,kk} \cdot y^k \right) \quad \forall kk \notin RAW(kk) \tag{5.14}$$

while with respect to the downstream selection is given as:

$$y^{k} \leq \sum_{kk} \left(\zeta_P^{k,kk} \cdot y^{kk} \right) \quad \forall k \notin PROD(k) \tag{5.15}$$

Domain Partitioning

The domain partitioning strategy consists of partitioning the domain of the continuous variables appearing in the bi-linear term, and use piecewise linear under- and overestimators over each partition. The domain partitioning can be mono-dimensional (when one of the variables appearing in the bi-linear term is partitioned) or bi-dimensional (when both are partitioned) (Bergamini *et al.*, 2005). While providing a tighter relaxation, this strategy causes an increase in the number of binary variables due to the piecewise linearization. Typically, it has been observed for similar problems that a partitioning over 2-3 intervals represents the best trade-off in terms of computational time (Karuppiah and Grossmann, 2006).
The McCormick relaxation for a bi-dimensional partitioned domain over a constant grid is given as:

$$F_f^{i,k,kk} \geq \sum_{o,p} \left(\Gamma^o \cdot SM_{D_f}^{i,k,kk,o,p} + F_{D_{out,f}}^{i,k,kk,o,p} \cdot \Lambda^p - \Gamma^o \cdot \Lambda^p \cdot w_f^{i,k,kk,o,p} \right) \tag{5.16}$$

$$F_f^{i,k,kk} \geq \sum_{o,p} \left(\Gamma^{o+1} \cdot SM_{D_f}^{i,k,kk,o,p} + F_{D_{out,f}}^{i,k,kk,o,p} \cdot \Lambda^{p+1} - \Gamma^{o+1} \cdot \Lambda^{p+1} \cdot w_f^{i,k,kk,o,p} \right) \tag{5.17}$$

$$F_f^{i,k,kk} \leq \sum_{o,p} \left(\Gamma^{o+1} \cdot SM_{D_f}^{i,k,kk,o,p} + F_{D_{out,f}}^{i,k,kk,o,p} \cdot \Lambda^{p} - \Gamma^{o+1} \cdot \Lambda^{p} \cdot w_f^{i,k,kk,o,p} \right) \tag{5.18}$$

$$F_f^{i,k,kk} \leq \sum_{o,p} \left(\Gamma^{o} \cdot SM_{D_f}^{i,k,kk,o,p} + F_{D_{out,f}}^{i,k,kk,o,p} \cdot \Lambda^{p+1} - \Gamma^{o} \cdot \Lambda^{p+1} \cdot w_f^{i,k,kk,o,p} \right) \tag{5.19}$$

where o is the index of partition for the flow variable F and p is the index indicating of partition over the split variable SM, over which the disaggregated variables $F_{D_{out,f}}^{i,k,kk,o,p}$ and $SM_{D_f}^{i,k,kk,o,p}$ are defined; Γ^o and Λ^p indicate the grid for partitioning of the

variables F and SM respectively, and $w_f^{i,k,kk,o,p}$ is a binary variable indicating the active partition o, p for the bi-linear term i, k, kk with respect to the outlet f.
The corresponding convex hull is given as:

$$\Gamma^o \cdot w^{i,k,kk,o,p} \leq F_{D_{out,f}}^{i,k,kk,o,p} \leq \Gamma^{o+1} \cdot w^{i,k,kk,o,p} \tag{5.20}$$

$$\Lambda^p \cdot w^{i,k,kk,o,p} \leq SM_{D_f}^{i,k,kk,o,p} \leq \Lambda^{p+1} \cdot w^{i,k,kk,o,p} \tag{5.21}$$

$$\sum_{o,p} \left(w^{i,k,kk,o,p}\right) \leq 1 \tag{5.22}$$

$$F_{out1}^{i,k} = \sum_{o,p} \left(F_{D_{out,f}}^{i,k,kk,o,p}\right) \quad \forall kk \tag{5.23}$$

$$SM_f^{k,kk} = \sum_{o,p} \left(SM_{D_f}^{i,k,kk,o,p}\right) \quad \forall i \tag{5.24}$$

Multi-level decomposition

In order to mitigate the increase of complexity due to the definition of additional binary variables associated to the domain partitioning, recent studies suggested the use of multi-level domain partitioning strategy (Bogataj and Kravanja, 2012). The main features of the multi-level domain partitioning is obtaining a progressive tightening of the lower bound over the progression of the algorithm, through multiple solution of the relaxed sub-problem on an augmented partitioning grid.
The algorithm for the solution of non-convex MINLP through multi-level domain partitioning and McCormick relaxation is shown in figure 5.4, for two levels of relaxation. The algorithm starts with the solution of the relaxed problem, based on a predefined domain partitioning grid. As a solution, an upper bound (in case of maximization) is obtained, together with the active partition for the variables appearing in the bi-linear term (indicated in yellow in figure 5.4).
The binary variables are fixed and the local problem is solved, starting from the solution of the relaxed problem. If the termination criteria are met, the procedure is stopped and the lower bound is reported as the solution.
If the termination criteria are not met, the algorithm proceeds with the second level of relaxation. The active portions of the grid are partitioned, and an augmented grid is defined. As shown in figure 5.4, the augmented grid is build in order to increase the resolution in the portion of the search space where the optimal solution is supposed to be found (based on the first level relaxation), while it is kept coarse in the portions of space where it is not supposed to be found. In this way, the computational complexity associated with the increase of resolution is managed, allowing to identify a better trade-off between computational cost and relaxation tightness.
The second level relaxation is solved, obtaining a new upper bound. Being the second level relaxation tighter, the second level upper bound is less or equal to the first level one, and therefore the optimality gap is reduced.
The solution of the second level relaxation is fixed and the local problem is solved. If the termination criteria are met, the algorithm is stopped and the optimal lower

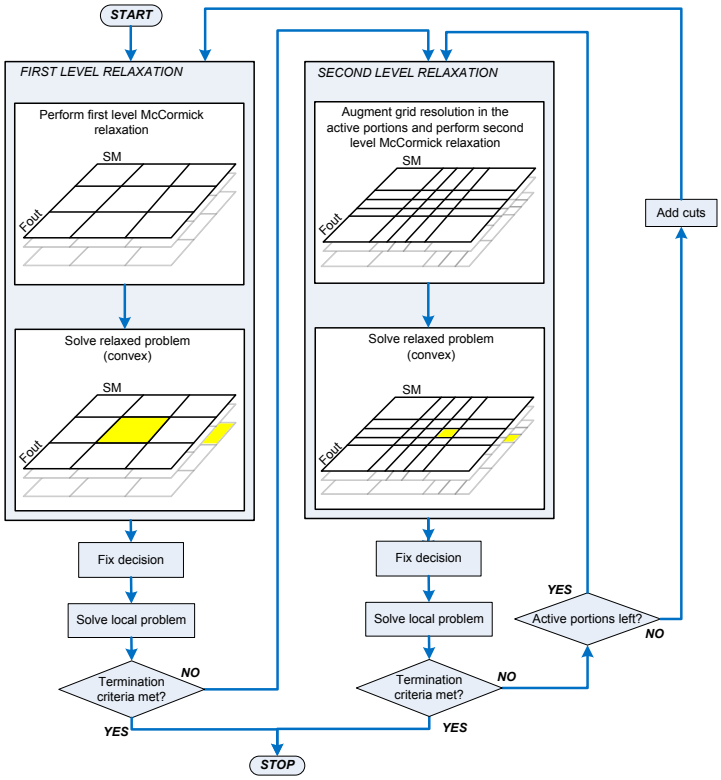


Figure 5.4. Algorithm for multi-level decomposition, based on two level of domain partitioning (Bogataj and Kravanja, 2012)

bound is reported as the solution. Otherwise, if any active portion is left the augmented grid is expanded to these portions, and the second level relaxation is iterated. If no active portion is left, cuts are added to exclude the obtained topology from the solution space, and the overall procedure is iterated.
Although in principle such a multi-level domain partitioning could be executed over more than two levels, the increase in relaxation tightness does not appear to compensate the additional algorithmic complexity associated to higher order levels. Therefore, the framework integrates a two-level domain partitioning algorithm.

Other strategies

As an alternative, the use of a parametric disaggregation strategy to limit the number of additional binary variables resulting from the piecewise relaxation has recently been proposed (Teles *et al.*, 2012). At the current status, parametric disaggregation is not implemented in the framework, but it could be integrated as a further development.

5.3 Superstructure reduction policy

As previously explained, decision-making under uncertainty problems are solved through sampling-based methods. This method results in the increase of problem size, which may cause the problem to become difficult to solve.
Because of the incremental structure of the workflow integrated in our framework, a large amount of information and results are generated prior to the formulation of the problem under uncertainty. This information and intermediate results can be used in order to facilitate the solution of the later and more complex steps of the methodology. To this goal, a superstructure reduction policy has been developed and integrated in the framework. The main feature of this policy is to make use of information and intermediate results generated prior to the formulation of the two stage stochastic problem (step 5 of the workflow) as a rationale for the simplification of large complex problems, to a solvable form.
The superstructure reduction policy is based on the elimination of process intervals which have not been selected in any of the uncertainty mapping solutions. As a result, the problem under uncertainty is formulated and solved for a smaller superstructure, which implies a reduction in terms of problem size (number of constraints and binary variables) and of search space, and consequently of computational resources necessary for the solution of the problem.
It should be underlined that the superstructure reduction policy is a simplification scheme. For a general optimization problem, the superstructure reduction policy is not an exact method, and the solution of the simplified problem may not be the global optimum of the original problem. As a consequence, this policy should be applied only for problems which are otherwise not tractable.

Remark 1 *Under the condition of i) problem linearity, ii) uniform probability distribution of the uncertain data, iii) feasibility and optimality of the same base for every network over the uncertain domain; the proposed superstructure reduction policy is an exact method, and the solution of the simplified problem is the global optimum solution of the original stochastic problem.*

Remark 1 states that, under the given conditions, if $y*$ is the optimal network under uncertainty, then a point in the uncertain domain $\Theta_o \in [\Theta^{LO}; \Theta^{UP}]$ exists, so that $y*$ is the optimal network at Θ_o.
In mathematical terms the hypothesis is:

$$\forall y* \;\; : \;\; y* \in \{0,1\}^m; \;\; \max_{x,y} E_\Theta(f(x,y,\Theta)) = E_\Theta(f(x,y*,\Theta)) \tag{5.25}$$

and the thesis is:

$$\exists \Theta_o \;\; : \;\; \Theta_o \in [\Theta^{LO}; \Theta^{UP}]; \;\; f(x,y*,\Theta_o) = \max_{x,y} E_\Theta(f(x,y,\Theta)) \tag{5.26}$$

Proof Consider a stochastic MILP problem, with one uncertain parameter $\theta \in [\theta^{LO}; \theta^{UP}]$. For each of the n networks identified by performing the procedure described as uncertainty mapping (step 4 of the framework), a stochastic LP problem is defined.

According to the Basic Sensitivity Theorem (Fiacco and Bank, 1984), if the same base is feasible and optimal over the uncertain space, a linear relationship exists between the value of the uncertain parameter θ and the value of the objective function value f of the LP problem.

$$f(\theta) = f(\theta_o) + c(\theta - \theta_o) \tag{5.27}$$

Where c is a constant term (often referred to as marginal value). For more details on equation 5.27 and the basic sensitivity theorem, including the proof, the reader is referred to Fiacco and Bank (1984).
The expected value of the objective function for each network is calculated as:

$$E_\theta(f(\theta)) = \int_{\theta^{LO}}^{\theta^{UP}} (f(\theta) \cdot P(\theta) d\theta) \tag{5.28}$$

Where $P(\theta)$ is the probability function of θ. For uniform probability distribution and linear function $f(\theta)$ this is :

$$E_\theta(f(\theta)) = f(\theta_{MEAN}) \tag{5.29}$$

Which proves the observation, when $\theta_o = \theta_{MEAN}$. The same reasoning can be replicated for more than one uncertain data.

Note 3 *The conditions under which this proof is obtained are quite restrictive, and correspond to particular cases in which, as shown by equation 5.29, the solution under uncertainty is equivalent to the deterministic solution, obtained for the expected value of the uncertain data.*

5.4 Variable initialization

In order to facilitate the solution of large and complex problems, the framework employs a variable initialization method, based on the reuse of previous results (if available) as initial values for the optimization variables. In particular, within the incremental structure of the workflow introduced in the previous chapter, solutions of the earlier steps are used as starting point in later steps.
The solution of the deterministic problem (step 3) is in fact employed for the initialization of the variables in the uncertainty mapping analysis (step 4). Similarly, the uncertainty mapping results are employed to initialize the variables for the two stage stochastic programming problem formulated in step 5 and 6.
Depending on the problem, it has been observed that such a simple initialization procedure can allow a considerable reduction in computational time for the solution. The initialization is therefore executed as a preliminary step for all solutions.

6

Supporting methods and tools

In this chapter, the methods and tools developed to support the workflow of problem formulation and solution presented in chapter 4 are introduced.

6.1 Superstructure synthesis method

As explained in chapter 4, in the first step of the problem formulation workflow the user is requested to specify the search space for the optimization-based design problem, and to represent it in the form of a superstructure. As described in section 4.1.2, this task may present a certain degree of complexity, and different methods exist for its execution.
In order to facilitate this task, an insight-based method for superstructure synthesis has been developed based on a modification of the systematic process synthesis methodology proposed by Siirola (1996). Through this method, the search space for the design problem and its superstructure representation are generated in a systematic manner, based on the engineering, economic and regulatory knowledge available with respect to the specific problem. The integration of the superstructure synthesis method within the framework structure is realized in the form of a guidance document, as outlined below.
The method, whose schematic representation is reported in figure 6.1, is composed by 2 main sections: a process steps section and a process intervals section. The process steps section guides the user through the systematic generation of the process steps contained in the superstructure. The procedure is based on *means-ends* analysis, a problem-solving strategy developed for artificial intelligence, aiming at the systematic identification of *means* (in this application represented by process steps) suitable to achieve a predefined *end* (the conversion of raw materials into products) (Simon, 1981).
In the intervals section, process intervals that are suitable for the execution of the identified process steps are added to the superstructure.
The superstructure generation method is briefly described in the next sections. In the second case study, the features of the method will be demonstrated by applying it to the synthesis of the superstructure for soybean processing.

6.1.1 Process steps section

Step 1: Problem definition

The problem is defined by stating goals and objectives, as well as by defining its scope, with respect to raw materials, products and process technologies.

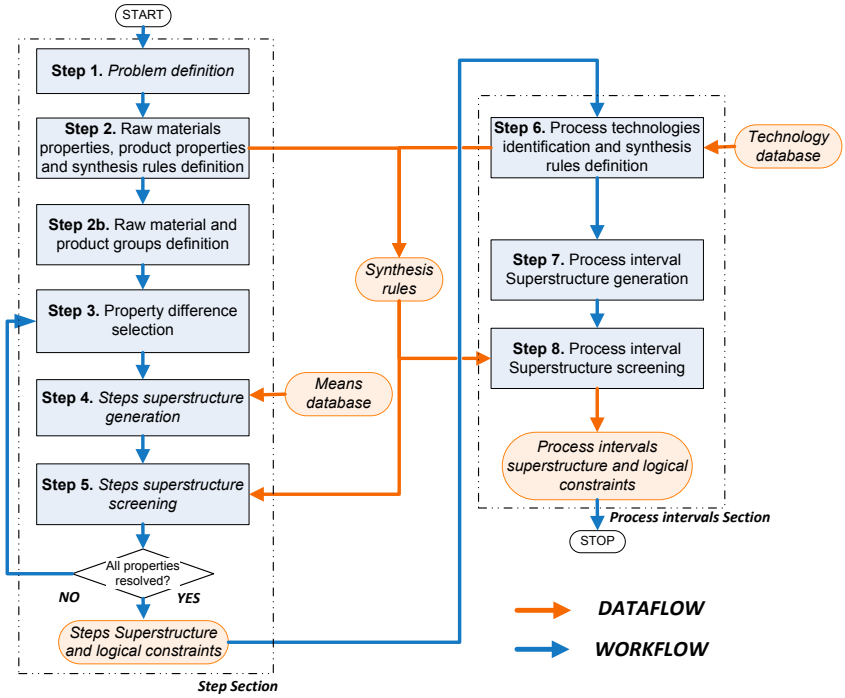


Figure 6.1. Schematic representation of the superstructure generation method

Step 2: Raw material and product properties and synthesis rules definition

All possible raw material and product alternatives are identified, and their specifications in terms of properties (i.e. composition, temperature, phase etc.) are defined. Commercial and engineering insights are converted into synthesis rules, which establish conditions that the superstructure needs to satisfy. Those rules can be classified as:

- product rules
- process rules
- process-product rules

Product related rules establish logical conditions for the selection of products, and are in general originating from commercial considerations. An example of product related rule is *"product P_1 has to always be produced"*, stating that out of all possible product alternatives, one (product P_1) needs to always be produced, and therefore implying that configurations not yielding the production of P_1 should be excluded from the search space.
Process related rules identify process steps which are technically impossible or undesirable (because of cost, complexity, freedom to operate, regulatory reasons, etc), and are in general derived from engineering insights and regulations. Example of

process rules are *"separation of x from y is impossible"* or *"separation of x from y can be done only after separation of z"*.
Process-product rules establish relations between a certain product and the process which has been used for its manufacturing, and are in general derived from regulations. An example is *"product P_1 can be produced only through a solvent-free process"*. Through the definition of the synthesis rules, the available knowledge with respect to the technological, commercial and regulatory aspects of the problem is collected and systematized in a structure, which allows its use in the formulation of the design problem.

Step 3: Property difference definition and selection

Raw material and product properties defined in the previous step are compared, and a table of property differences is compiled.
From an economic perspective, these differences in properties represent the cause of the difference in value between raw materials and products. The goal of the process is therefore the elimination of those property differences, in order to extract the associated value. The identified differences are ordered according to the property they are related to. To this end, the hierarchical approach proposed by Siirola (1996) is used, and property differences with respect to species identity are placed first, followed by differences in amount, composition, temperature, pressure, size and geometry and finally geographical location.
In the next steps of the method, each of these differences will be resolved through an iterative process, proceeding according to the order defined.

Step 4: Process steps superstructure generation

In this step, the first unresolved property difference among the list compiled in the previous step is considered. The knowledge base is searched, and the means (process steps) suitable for the elimination of the property difference considered are retrieved from a means database. An example of the knowledge base structure is reported in table 6.1.
The identified process steps are added to the superstructure, in all possible configurations. The property difference which has been resolved is eliminated from the difference list.

Table 6.1. An example of knowledge base for means-ends analysis (note that the process steps list is not exhaustive)

property difference	**process step**
species identity	reaction
amount	dosage, flow splitting
concentration	separation
phase	vaporization, condensation
temperature	heating, cooling
pressure	compression, expansion
size and geometry	agglomeration, milling,..
location	transportation

Step 5: Process steps superstructure screening

The synthesis rules defined in step 2 are used to identify non-legal configurations (meaning configurations violating one or more synthesis rules), which are eliminated from the superstructure. The elimination of non-legal alternatives is performed by eliminating a process step or a connection from the superstructure, or by defining logical constraints, similar to the ones described in section 3.4.1.

Steps 4-6 are repeated until all property differences have been addressed, and the process steps superstructure and the corresponding logical constraints have been defined.

Incremental superstructure synthesis

Note 4 *The execution of the method may become cumbersome when a large number of raw materials and products are considered, because of the combinatorial explosion of the number of alternatives which need to be generated, screened against rules and systematized in the superstructure.*

In order to cope with the problem highlighted in note 4, the process step section of the superstructure synthesis method can be executed in an incremental manner, based on an iterative execution of the synthesis procedure.
In the first iteration, raw materials and products are lumped into coarse raw materials and products groups (by grouping together similar raw materials and products), resulting in a simplified version of the synthesis problem. At each successive iteration, the superstructure obtained in the previous iteration is refined by considering finer groups, until the original raw material and product list is considered. The main advantage of this procedure is its ability to eliminate some of the non-legal configurations in the early iterations (on a relatively small and simple superstructure), and so to simplify the execution of later iterations, which may be characterized by large superstructures.
The incremental synthesis procedure is described in figure 6.2. With respect to the above described methodology, the incremental procedure requires an additional step for the definition of raw materials and products groups (step 2b). Steps 2b-5 are executed iteratively; at each iteration, progressively finer groups are defined in step 2b, and the superstructure obtained at previous iterations is modified, until the superstructure for the original raw material and product list is obtained.
An example of the application of this procedure is given in chapter 9, in which the iterative procedure is applied to manage the complexity associated to the existence of more than 20 product alternatives.

6.1.2 Process intervals section

In this section, intervals are added to perform each of the process steps identified in the process step section.

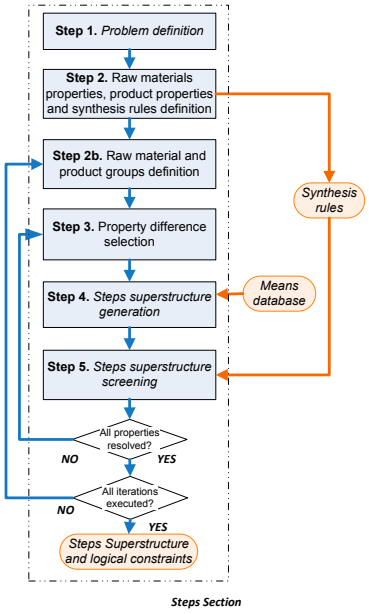


Figure 6.2. Iterative procedure for synthesis of large superstructure

Step 7: Process technologies identification and synthesis rules definition

The knowledge base is consulted, and the database of process intervals is searched to identify the ones which can execute each of the process steps identified. An example of the structure of such a knowledge base is reported in table 6.2.
Each of the process technologies considered is analyzed, and secondary input-output streams are identified. Examples of secondary inputs are utilities and chemicals required by each process interval, while secondary outputs include side-streams. All secondary inputs and outputs are added to the problem formulation, as a complement to the raw material and product lists. The scope of the synthesis problem is therefore expanded, by including the problem of sourcing those utilities and chemicals, and disposing or utilizing all side-streams. Typical options for sourcing utilities and chemicals are:

1. purchase from a supplier
2. on-site production
3. reuse of an existing side-stream

Similarly, management strategies for side-streams include:

1. valorization as co-product (with or without requiring additional processing)
2. disposal as waste (in general at the cost of a disposing price)
3. mixing with one of the existing products (with or without requiring additional processing)

4. reuse as utility or chemical in a process

As a result of this secondary flow analysis, new raw materials, products, process intervals and process steps may be defined.
Finally, synthesis rules are defined at process interval level, to exclude infeasible or undesired configuration from the search space. These rules complement the list defined in step 2 at process step level.

Table 6.2. Example of a process interval database

Step: Reaction				
Interval	**Main react.**	**Main prod.**	**Notes**	**Rules**
Conversion 1	A	C	high T	not applicable if thermal unstable compounds
Conversion 2	A	B		
Step: Separation				
Interval	**Mixture**	**Sep. method**	**Notes**	**Rules**
Separation 1	A,B,C	distillation		
Separation 2	A,B	extraction	organic solvent	not applicable for food

Step 8: Process intervals superstructure generation

Each of the process steps of the superstructure is populated by one or more process intervals, connected in all possible configurations.

Step 9: Process intervals superstructure screening

The alternatives generated in the previous step are screened against the synthesis rules. Non-legal configurations (violating one or more synthesis rules) are eliminated from the search space.
The elimination of those alternatives from the search space can be done either by removing a process interval or a connection from the superstructure, or by defining logical constraints between the selection of process intervals, as explained in section 3.4.1.
As a result of this screening, the superstructure is defined, along with a list of logical constraints. Such a superstructure, generated in a systematic manner, contains all possible configurations based on the available library of process intervals and process steps, while it does not contain alternatives which, based on the knowledge available, are known to be unfeasible or unwanted.
Furthermore, one of the advantages of the use of this methodology is that the explicit definition of the synthesis rules allows a transparent and exhaustive documentation of the conditions under which such a superstructure has been generated. This, as well as the use of a systematic procedure, facilitates the user in keeping an unbiased approach while considering all possible configurations, resulting in the definition of a comprehensive superstructure, which may contain innovative configurations.

6.2 Data architecture

As described in chapter 4, the formulation of the synthesis and design problem requires collecting and specifying a large number of multi-source data. As discussed, from a user perspective this often results in a frustrating and time consuming task, in which the risk of committing errors (and consequently compromising the quality of the obtained results) is in general quite elevated.
In order to facilitate this step, the framework integrates a data architecture, whose main feature is to provide a structure for the collection and management of the information and data associated to the problem.
In the next section, the data structure will be defined, with respect to the formulation of problems based on the generic process interval model described in chapter 3. A similar concept may be applied when different models are employed.

6.2.1 Multi-layer data structure

When the generic process interval model is used, the specification of the synthesis and design problem requires the specification of the data reported in table 6.3.

Table 6.3. Lower and upper bounds and default value for the problem data

Data	Elements	LO	UP	default	notes
$\phi^{i,k}$	raw material flow	0	$+\infty$	0	
$\alpha^{i,k}$	utility	0	1	0	
$\mu^{ut,i,k}$	utility	0	$+\infty$	0	
$\gamma^{i,k,rr}$	reaction	$-\infty$	$+\infty$	0	
$\theta^{i,k}$	reaction	0	$+\infty$	0	
$\sigma^{i,k}$	separation	0	1	1	
$\delta^{i,k}$	waste	0	1	0	
$\eta^{k,kk}$	transportation	0	$+\infty$	0	
$\zeta_f^{k,kk}$	superstructure	0	1	0	binary
$\nu^{st,k}$	superstructure	0	1	0	binary
π_R^k	ob. funct.	$-\infty$	$+\infty$	0	
$\pi_U^{i,k}$	ob. funct.	0	$+\infty$	0	
π_P^k	ob. funct.	$-\infty$	$+\infty$	0	
$\pi_T^{k,kk}$	ob. funct.	0	$+\infty$	0	
π_C^k	ob. funct.	0	$+\infty$	0	
$F_{IN}^{i,kk^{UP}}$	limits	0	$+\infty$	0	
$F_{IN}^{i,kk^{LO}}$	limits	0	$+\infty$	0	

Within the architecture integrated in the framework, these data are organized in 7 data layers. The resulting structure is shown in figure 6.3, where the 7 data layers are shown in the first column, examples of the data contained in each layer constitutes the second column, and the data source (in terms of expertise) is shown in the third column. In the following paragraphs, the layer structure will be briefly described.

Data related to *alternatives* are organized in the first layer, which contains the list of process intervals, steps, components and reactions which are considered in

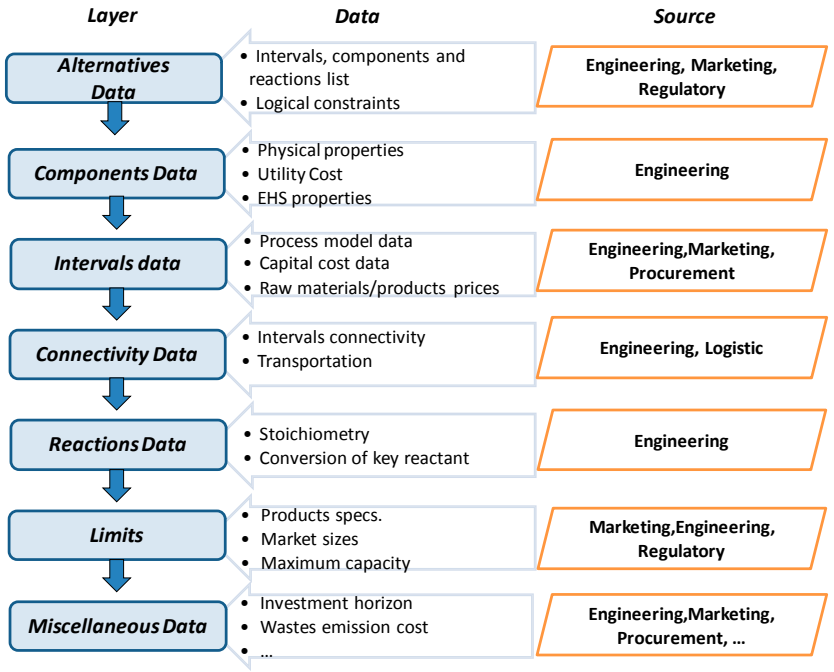


Figure 6.3. Layer decomposition of data architecture

the problem. These data define the search space of the design problem, and set its dimension in term of number of equations and data needed for its formulation. For example, for a network problem in its standard formulation, the number of equations and data can be calculated as a function of the first layer data as:

$$N_{equations} = 2 + N_k + 12N_i \cdot N_k + 2N_i \cdot N_k^2 \tag{6.1}$$

$$N_{data} = 2{+}4N_i{+}6N_k{+}N_{rr}{+}N_{st}{+}2N_k^2{+}N_k{\cdot}N_{st}{+}5N_i{\cdot}N_k{+}N_i^2N_k{+}N_i{\cdot}N_k\cdot N_{rr} \tag{6.2}$$

where $N_{equations}$ and N_{data} are the number of problem equations and data respectively, N_i is the number of components, N_k of process intervals, N_{st} of process steps and N_{rr} the number of reactions.

Component data constitute the second layer, in which all data related to the characterization of process and utility components (such as physical properties and specific utility price) are contained.

The third layer is constituted by process *interval* data. These include the data for the calculation of the mass balance (such as specific utility consumption, split factors for separation, etc.), raw materials composition and price, products price and capital cost data.

Interval *connectivity* data are included in the fourth layer. These include the definition of the possible connections between process intervals existing in the superstructure, the allocation of primary and secondary outlets for each interval and the transportation costs.

In the fifth layer, data related to the definition of *reactions* such as stoichiometry and conversions are grouped.
The sixth layer contains data related to *limits*, such as product specifications (in composition and flow), maximum throughput for each process interval, etc.
The seventh layer contains *miscellaneous* data used in the constraints or in the objective function (such as investment horizon, discount rate, etc.).

6.2.2 Relations between data layers

The main feature of the above described layer-based data architecture is its hierarchical structure. Data layers are organized in a way such that the specification of a data in an upper layer conditions data in subsequent lower layers. These conditions are due to relations existing between the different layers, which can be classified as:

- dimensional relations
- value relations
- value conditioning relations

Dimensional relations are observed when the specification of a data defines the dimension (in term of matrix representation) of a data in a lower layer. An example of dimensional relations can be seen in the definition of the component list (layer 1) which sets the dimension of the vector of molecular weights (layer 2).
Value relations are observed when the specification of a data defines the value that an other data (in a lower layer) has to assume. An example of value relations is that the allocation of an interval to the product step (layer 3) implies that no reaction can occur in that interval, and therefore automatically sets the value of reaction conversion in that interval to zero (layer 5).
Value conditioning relations are observed when the value of a data is conditioned by one or more data specified in upper layers. An example of value conditioning can be observed considering that, once the component molecular weights are defined (layer 2), the law of mass conservation establishes a condition that reaction stoichiometry data (layer 5) need to satisfy for the specified reaction to be legal.
Within the framework, the hierarchical nature of the data architecture and the above described relations between layers are exploited to optimize the problem formulation workflow and enable its automation. The automation is realized through the development of a software tool, which serves as user-interface and guides the user through the specification of each data layer in a sequential manner. The software tool will be described in the next chapter.

Automatic data specification based on value relations

Value relations between data layers are exploited in order to automatically define the value of data in the lower layers, based on the value of data specified in the upper layers. Through this automatic specification, the number of data values that

the user needs to specify is reduced. As a consequence, the time needed for the formulation of the problem, as well as the probability of error in the data specification procedure is also reduced.
A summary of the value relations which are employed for automatic data specification is shown in table 6.4. The logic for the automatic data specification is that, if the condition reported in the first column of the table is respected, the data assignment reported in the corresponding right column is implemented. An example of the application of the logic is the following: if the separation elements for a given process interval is inactive (i.e. the condition reported in the seventh row of the table is met), then it is automatically specified that such a process interval will not have any secondary outlet connection (through the specification of the corresponding value of $\zeta_S^{k,kk}$).

Table 6.4. Automatic data specification

Condition	Data specification
$i \notin ut(i)$	$\pi_U^{i,k} = 0 \quad \forall k$
$k \notin RAW(k)$	$\pi_R^k = 0$
$k \notin PROD(k)$	$\pi_P^k = 0$
$k \in RAW(k) \cup PROD(k)$	$\alpha^{i,k} = 0 \quad \forall k$ $\mu^{ut,i,k} = 0 \quad \forall k$ $\gamma^{i,k,rr} = 0 \quad \forall k$ $\theta^{i,k} = 0 \quad \forall k$ $\sigma^{i,k} = 1 \quad \forall k$ $\delta^{i,k} = 0 \quad \forall k$
$kk \in RAW(kk)$	$\zeta_f^{k,kk} = 0 \quad \forall k$
$k \in PROD(k)$	$\zeta_f^{k,kk} = 0 \quad \forall kk$
$\sum_i \left(1 - \sigma^{i,k}\right) = 0$	$\zeta_S^{k,kk} = 0 \quad \forall kk$
$\zeta_P^{k,kk} = 1$	$\zeta_S^{k,kk} = 0 \quad \forall kk$
$\sum_f \left(\zeta_f^{k,kk}\right) = 0$	$\pi_P^{k,kk} = 0$

Coherence checks based on value conditioning relations

Value conditioning relations are used to define data coherence checks, which are enforced in order to identify non-legal or incoherent problem specifications. Through these checks, faulty problem specifications can be identified prior to the solution of the optimization problem, allowing to increase the reliability of the problem formulation workflow, while at the same time contributing to make its execution faster.
In particular, the coherence checks enforced are:

(a) Alternative internal coherence

(b) Network coherence

(c) Internal separation coherence

(d) Separators connectivity

(e) Connectivity internal coherence

(f) Reactant availability

(g) Reaction mass balance

(h) Reaction internal coherence

(i) Limits attainability

A schematic representation of those checks is reported in table 6.5, where each line represents the layer that is subject to the check, and the columns represents the input information for the execution of the check. Because of the above described hierarchical nature of the layer structure, table 6.5 is characterized by a lower triangular structure. Consequently, if each data layer is specified in a sequential manner, coherence checks can be executed in real time, allowing immediate identification of errors in the data specified by the user.

Table 6.5. Data specification coherence checks

	Alternatives	Components	Intervals	Connectivity	Reactions	Limits	Misc.
Alternatives	(a)						
Components							
Intervals	(b)		(b) (c)				
Connectivity			(d)	(e)			
Reactions	(f)	(g) (f)		(f)	(f)(h)		
Limits	(i)	(i)	(i)	(i)	(i)		
Misc.							

The *alternatives internal coherence* check (indicated as (a) in table 6.5) ensures that a legal declaration of alternatives is formulated, by checking that the number of process intervals is greater or equal to the number of process steps (including raw materials and products).
The *network coherence* verification (b) checks that at least one interval is associated to each of the process steps.
The *internal separation coherence* check (c) ensures that a legal declaration of the separation function is formulated. This check is given as:

$$\sum_{i} \left(\sigma^{i,k}\right) > 0 \quad \forall k \tag{6.3}$$

The *separation connectivity* check (d) verifies the coherence between the definition of the separation tasks and the process interval connectivity. In particular, the

check verifies that, if a process interval contains a separator, both a primary and a secondary outlet are defined for such an interval. This is done as:

$$\sum_{kk} \left(\zeta_S^{k,kk}\right) \geq \sigma^{i,k} \quad \forall i,k \tag{6.4}$$

The *connectivity internal coherence* check (e) analyzes the specification of the data defining the connection existing in the superstructure, to ensure that a coherent network is defined. To this end, the logic ensures that all intervals representing a processing step have at least one primary inlet and one outlet connection, that all raw materials have no inlet and at least one outlet; and that all products have at least one inlet and no outlet.

$$\sum_{f,kk} \left(\zeta_f^{k,kk}\right) \geq 1 \quad \forall k \notin PROD(k) \tag{6.5}$$

$$\sum_{k} \left(\zeta_P^{k,kk}\right) \geq 1 \quad \forall kk \notin RAW(kk) \tag{6.6}$$

$$\sum_{kk,f} \left(\zeta_f^{k,kk}\right) = 0 \quad \forall k \in PROD(k) \tag{6.7}$$

$$\sum_{k,f} \left(\zeta_f^{k,kk}\right) = 0 \quad \forall kk \in RAW(kk) \tag{6.8}$$

The *reactant availability* (f) test verifies the reactants specified for a given reaction can exist in the reactor inlet. This check is given as:

$$\sum_{k,f} \left(F_{out_f}^{i,k^{UP}} \cdot \zeta_f^{k,kk}\right) \cdot \left((1 + \alpha^{i,kk} \cdot \mu^{i,kk})\right) \geq 0 \qquad \forall \{i;kk\} \ : \ \gamma^{i,kk} \leq 0 \tag{6.9}$$

where $F_{out_f}^{i,k^{UP}}$ is calculated according to equations 5.10 - 5.13. It should be underlined that this check performs the verification with respect to the availability of one reactant at the time, without considering the limitation to selection of process intervals given by the logical constraints. Therefore, this check represents a necessary (and not sufficient) condition of the feasibility of the reaction specified.
The *reaction mass balance* check (g) ensures that reactions are specified in compliance with mass conservation law. This check make use of the molecular weight defined in the components layer to verify the coherence of the reaction stoichiometry data $\gamma^{i,k,rr}$, as:

$$\sum_{i} \left(\gamma^{i,k,rr} \cdot MW^{i}\right) = 0 \quad \forall rr;k \tag{6.10}$$

The *reaction internal coherence* verification (h) checks the correct definition of reaction stoichiometry, by verifying that at least one reactant and one product are specified, and that the key reactant is selected coherently with respect to the stoichiometry.
The *limits attainability* verification (i) aims at verifying that a configuration exist, which allows to attain the defined limits. Depending on the type of limit enforced, different formulations of the check can be derived. As an example, if the limit

is given with respect to value of $F_{in}^{i,k^{LO}}$ (meaning the minimum flow of a certain component i in the inlet of interval k, the check is given as:

$$\sum_{k,f} \left(F_{out_f}^{i,k^{UP}} \cdot \zeta_f^{k,kk} \right) \geq F_{in}^{i,k^{LO}} \tag{6.11}$$

where $F_{out_f}^{i,k^{UP}}$ is calculated according to equations 5.10 - 5.13. As for check (f), it should be underlined that this check performs the verification with respect to the attainability of one limit at the time, and without considering the limitations on network configuration imposed by logical constraints. As a consequence, it constitutes a necessary (but not sufficient) condition for the attainability of the limits specified.

6.3 Wastewater characterization

As described in the introduction, one of the relevant applications of optimization-based design is related to wastewater treatment and reuse networks. Such an application presents some specific challenges, which are related to the complexity of the chemical systems and processes typical of wastewater treatment engineering.
Among others, one of the challenges is related to the characterization of the contaminants contained in wastewater streams. A wastewater stream (whether originated from an industrial or a domestic source) will typically contain a very large number of chemical species, which are considered contaminants. Because of their number, it is often impractical or impossible to track the composition of each of the contaminant species; therefore it is common practice in the wastewater engineering community to characterize water composition with respect to pseudo-components (Tchobanoglous *et al.*, 2003). In wastewater practice, the most followed approaches for contaminants characterization are based on:

- traditional wastewater pseudo-components
- Activated Sludge Model (ASM) components

The *traditional* approach employs pseudo-components, based on first-principles. As an example of this method, it can be considered how many organic compounds present in wastewater (including protein, fatty acids, amino acids, etc.) are lumped through the definition of Chemical Oxigen Demand (COD), a measure of the electron donor capacity of a given wastewater (Roels, 1983). Other examples include the Biochemical Oxygen Demand (BOD), Total Kjeldahl Nitrogen (TKN), etc.
Despite having been widely used, this characterization is often too simplistic and disregards important differences in wastewater composition especially with regards to organic carbonaceous matter (Henze *et al.*, 1993b, 2002). Moreover, since a given contaminant may be detected by more than one analytical test, those pseudo-components are not independent. This feature complicates the definition of mass conservation relations (which have to be written in order to take into account those redundancies), and therefore makes this characterization method not suitable for modeling purposes.
The use of *Activated Sludge Model (ASM) components* represents an alternative for

more detailed wastewater characterization. This method is based on 13 independent pseudo-components, defined as reported in table 6.6.

Table 6.6. ASM1 compounds (Henze *et al.*, 2002)

X_i	Inert particulate organic matter
S_o	Dissolved Oxygen
X_s	Slowly biodegradable substrate
S_{NO}	Nitrate
X_H	Heterotrophic biomass
S_{NH}	Ammonia
X_A	Autotrophic biomass
S_{ND}	Soluble biodegradable organic nitrogen
X_p	Inert particulate from biomass death
X_{ND}	Particulate degradable organic nitrogen
S_i	Inert soluble organic matter
S_{ALK}	Alkalinity
S_s	Readily biodegradable substrate

Within the wastewater community, the ASM wastewater characterization is considered a breakthrough achievement in the attempt to develop reliable mechanistic modeling of wastewater treatment systems, which was pioneered by the release of ASM1 (Activated Sludge Model No 1) by Henze *et al.* (1987). Since then this principle of wastewater characterization has been extended to cover many wastewater treatment aspects, and now constitute the basis of commercial process simulators used in wastewater treatment engineering (e.g. Biowin, GPSx, WEST, etc.) and commonly used by wastewater professionals (Sin *et al.*, 2005).
Because of the higher level of detail that it can offer, as well as of its increasing acceptance and use in wastewater treatment process simulators, the use of ASM components appears as the most suitable characterization method for optimization-based design of water and wastewater networks. Consequently, the ASM component characterization has been integrated within the framework.
Nevertheless, depending on the source and composition of the wastewater to be treated, different characterization methods may be more convenient. In some cases, in fact, the representation of the full spectrum of ASM components may be not required, or result in a too complicated problem formulation. In other cases (especially when the treatment of industrial wastewater is concerned, the ASM characterization may not contain all the contaminants that need to be tracked.
In order to accommodate such cases, the ASM component list can be modified, in order to disregard some of the components which are not needed. Similarly, it can be expanded to consider other pollutants, such as for example metals (Cr^{6+}, Cr^{3+}, Fe^{2+}, Fe^{3+}), gases (H_2S, CO, CH_4), etc.

Conversion between wastewater characterization methods

As mentioned before, although the ASM characterization is more convenient for modeling purposes, the use of traditional parameters such as COD and BOD still constitutes a standard within wastewater engineering. Consequently, conversion

approaches are needed, in order to translate measurements between the 2 characterization methods.
To this end, a pragmatic approach that relies on physical-chemical characterization of wastewater adopted from BIOMATH and STOWA protocols is used (Sin *et al.*, 2005). This approach relies on pragmatic conversion relations, such as the ones reported in equations 6.12 - 6.16 (in which the coefficients are referred to municipal wastewater).
If needed, different characterization methods can be employed, as reviewed in Sin *et al.* (2005).

$$COD = X_i + X_s + S_i + S_s + X_H + X_B \tag{6.12}$$

$$BOD = X_s + S_s \tag{6.13}$$

$$Oil\&Grease = 0.9 \cdot (X_s + S_s) \tag{6.14}$$

$$TSS = 0.75 \cdot (X_i + X_s + X_A + X_H + X_p) \tag{6.15}$$

$$TKN = (X_i + X_s + S_i + S_s) \cdot 0.05 \tag{6.16}$$

where TSS is Total Suspended Solids and TKN is Total Kjeldahl Nitrogen.

7

Software implementation

In this chapter, the software implementation of the framework described in the previous chapters is presented. First EOLO, a software developed to facilitate the task of data specification, is presented. Then, the practical implementation of the optimization problems in GAMS is briefly discussed. Finally, the implementation of bi-level solution methods is described.

7.1 EOLO

In order to automate some of the tasks of problem specification, the software program EOLO has been developed in the C# environment.
EOLO is a user interface for the formulation of processing network problems based on the generic process interval model, which has been designed according to the hierarchical data structure presented in the previous section. The software employs the relations between data layers (described in section 6.2.2) in order to facilitate the task of data specification, integrating the functions of automatic specification and data consistency checks.
Through these features, EOLO allows reducing the time needed for the formulation of a network problem, while at the same time increasing the reliability of problem specification.
Furthermore, EOLO integrates an excel-based database, where problem formulations can be saved, hence allowing storage and systematization of information in a library of raw materials, processes and products data.
A schematic representation of the use of EOLO for the formulation of different classes of problems is shown in figure 7.1. The user specifies the data by compiling data tables (with the support of the automatic data specification and coherence check features), or retrieving them from the excel database. Once the problem formulation is completed, EOLO generates a GAMS-readable binary file containing all problem data, and a GAMS file containing all problem equations, corresponding to a deterministic single-stream formulation. When the generated file is executed, GAMS automatically load the data from the binary file, solve the optimization problem and generates an excel file containing the results.
Consequently, standard processing network problems can be formulated and solved through the user interface of EOLO, without requiring any additional programming. This allows reducing the time needed for the formulation of this class of problem, as well as broadening the range of potential users of this optimization-based design method, by making it accessible to professionals who are not experts in formulating and solving optimization problems in GAMS.
At the present stage of development, EOLO does not support the automatic formu-

lation of problems under uncertainty or multi-stream. This feature is considered as one of the useful areas for further development (see section 11.4.1).
Nevertheless, the standard problem formulation generated by EOLO represents a good starting point, that expert users can modify in order to obtain these more complex problem formulations. In this way, the amount of work required for the formulation of more complex problems is also reduced.

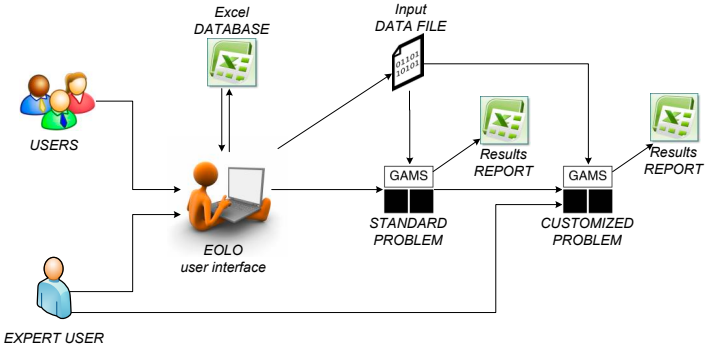


Figure 7.1. Schematic representation of the use of EOLO for problem formulation

7.1.1 Formulation of network problems through EOLO

In this section, the execution of the workflow for problem formulation through EOLO will be presented, by describing each of the components integrated in the software.

Selection of problem workflow

As a first step, the user is requested to specify if he/she intends to formulate a new problem or to load an existing formulation from the problem database. Furthermore, the class of problem which needs to be formulated is specified at this stage, by selecting among deterministic and stochastic formulation, and specifying whether a single- or multi-stream solution is required. As previously explained, at the present stage EOLO implementation is completed for the formulation of deterministic single-stream problem only, therefore only this option is currently available.

Layer 1: alternatives

In the first layer, data related to alternatives are specified by defining the number of process steps, process intervals, components, utilities and reaction. Moreover, intervals can be pre-allocated to the different steps.
A screenshot of EOLO interface for alternative specifications is reported in figure 7.2.
As explained in the previous sections, data specified in this layer set the dimension of the vector and matrix for the lower layer data. This feature is exploited by EOLO, which uses the data specified in the first layer to set up the structure for the other data.

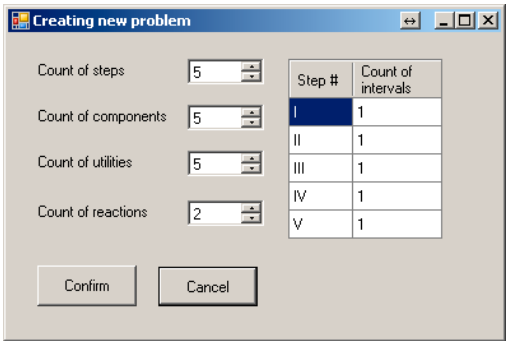


Figure 7.2. Problem Formulation Software: user interface to input alternatives data

Layer 2: components

The second step of the problem formulation workflow is the definition of the data related to the components. The component list is defined at this step, and among the list utility components are identified through a check box.
For each component, relevant physical properties are specified. Specific prices are defined for all the utility components, while other components prices are automatically set to zero, according to the logic defined in table 6.4. A screenshot of the components data layer tab in EOLO is reported in figure 7.3.

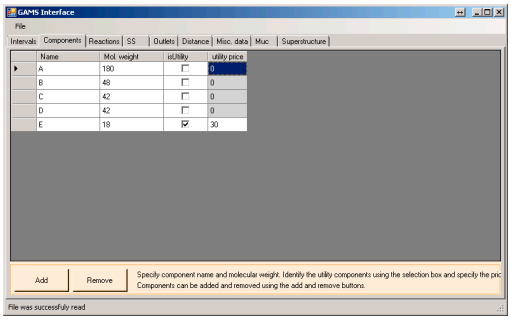


Figure 7.3. Problem Formulation Software: Components data

Layer 3: intervals

The third component of EOLO is related to the definition of data related to process intervals. Those include all data required for the formulation of the process model (with the exception of reaction data), as well as those required for the calculation of the investment cost. Moreover, raw material and product prices are specified in this step.
A screenshot of the alternative data layer tab in EOLO is reported in figure 7.4. The interval tab is divided in 3 areas. On the left hand-side, an overview of all process intervals and of their allocation to the different process steps constituting

the superstructure is shown. In this section, the user can edit the intervals name, as well as specify the cost function parameters associated to each of them. These include the raw material cost (for intervals allocated to the first step) and product prices (for intervals allocated to the last step). For the other intervals, those data are automatically fixed to zero according to the automatic specification logic described in table 6.4. Finally, the coefficients π^k_{Ca} and π^k_{Cb} for the capital cost calculation (equation 3.34) are specified in the column denominated p and q.
By clicking on the raw corresponding to a process interval, its data are displayed in the 2 tables on the right hand-side. In the bottom part, the user can define the value of the parameter $\mu^{ut,i}$, which is used to calculate the utility consumption through equation 3.2.
In the top table, for each interval allocated to step I the user can define the raw material composition ϕ^i, the data related to utility mixing α^i, separation σ^i and waste production δ^i (in the figure identified as phi, alpha, split and SW respectively), according to equations 3.3, 3.9 and 3.5.
The network coherence check and the separation coherence check (equation 6.3) described in the previous chapter are automatically executed while the data are specified, and warnings are issued when non legal specifications are defined by the user.
An overview of the process tasks (e.g. utility, separation, etc.) which, based on the specified data, are active for each process interval is displayed in the form of check boxes.

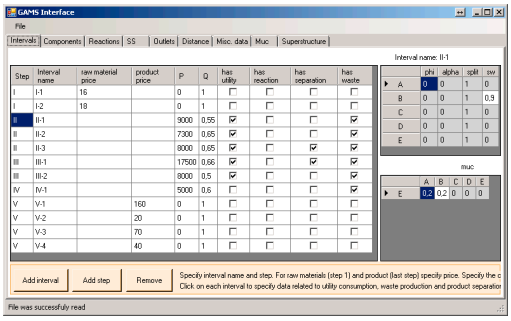


Figure 7.4. Problem Formulation Software: Intervals data

Layer 4: connectivity

In the fourth layer, connectivity data are specified, by defining the possible connections between the process intervals contained in the superstructure (see figure 7.5). In order to facilitate the specification of these data, for each source interval (represented as rows of the matrix $\zeta^{k,kk}_f$) the intervals allocated to the successive step are highlighted in green.
This visualization facilitates the definition of connection data, by clearly identifying recycle connections (appearing in the left hand of the green cells) and bypasses (appearing on the right side of the green cells). As an example, it can be seen that

the specification reported in figure 7.5 does not include any recycle stream; on the contrary, the connection between interval *III-2* and *V-3* bypasses a process step. Moreover, transportation data are also specified within this layer (see figure 7.6). To facilitate this task, since transportation distances can be specified only if connections are existing (table 6.4), EOLO highlights the existing connections in white for the user to input the data, and automatically fixes the other values to zero. The separation connectivity check (eq. 6.4) and the connectivity internal check (eq. 6.5-6.8) are also implemented automatically.

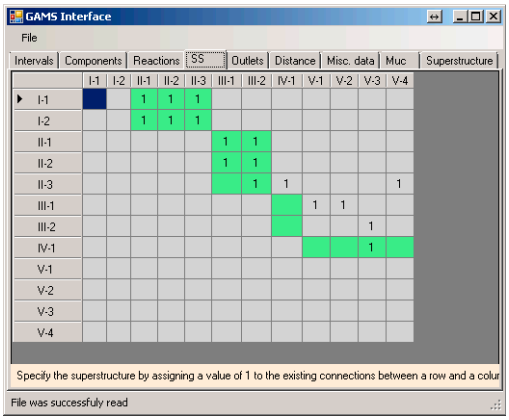


Figure 7.5. Problem Formulation Software: Connections data - connections

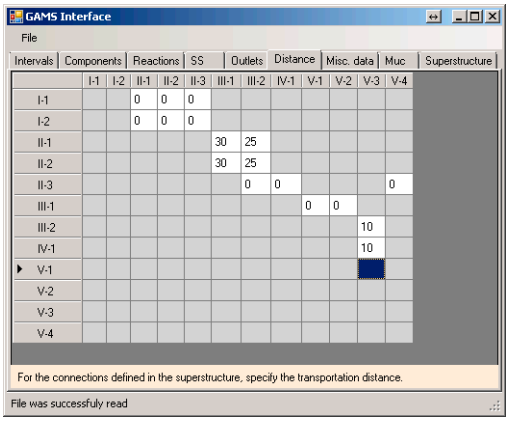


Figure 7.6. Problem Formulation Software: Connection data - transportation distance

Layer 5: reactions

The fifth data layer requires the specification of the reactions occurring in the processing network, which corresponds to the definition of stoichiometry and conversion for each of the reaction, as well as their allocation to process intervals.

A screenshot of the reaction data tab of EOLO is reported in figure 7.7. In the top part, for each of reaction (line), the stoichiometry is defined. The software verifies the reaction data that are specified by the user by checking the mass balance and the coherence of the defined stoichiometry. In the bottom part of the screen, for each reaction a key reactant is chosen, and the reaction conversion is defined for each process interval in which the reaction takes place.

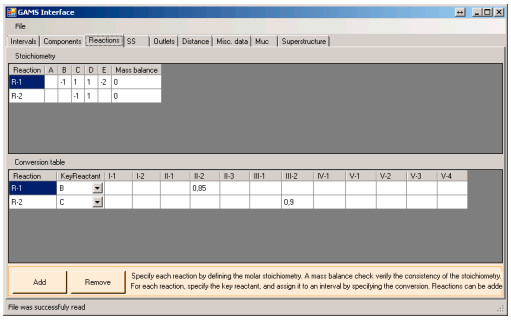


Figure 7.7. Problem Formulation Software: Reactions data

Layer 6: limits

The definition of limits constitutes the sixth layer of the data architecture. This step has not yet been implemented in EOLO, as well as the limit attainability test described in the section 6.2.

Layer 7: miscellanea

The last layer of the data structure is concerned with the definition of miscellaneous data. At the present stage of development of EOLO, this corresponds to the definition of penalty for waste emission, as well as to the definition of the data required for the setup of the piecewise linearization of the capital cost constraints, according to the bi-level decomposition scheme presented in section 5.2.1.
In particular, as highlighted in figure 7.8, the user can define the number of intervals for the linearization as well as the grid with respect to the flow variable (indicated as $\Gamma^{k,j}$ in equation 5.2).

Superstructure Visualization

When the problem specification steps are completed, the superstructure resulting from this formulation is visualized (see figure 7.9). When the problem specification is saved, the corresponding GAMS file is generated. As described, this file can be either executed in order to obtain the solution of the single-stream problem, or used as starting point for the development of more sophisticated problem formulation.

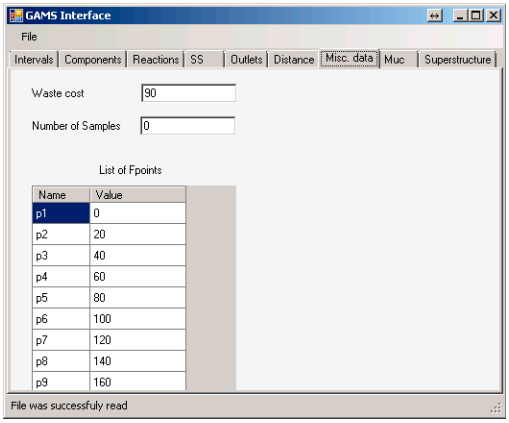


Figure 7.8. Problem Formulation Software: Miscellaneous data

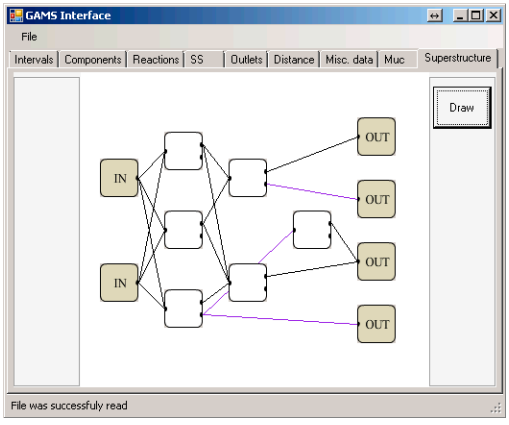


Figure 7.9. Problem Formulation Software: Superstructure

7.2 Optimization template files

In order to facilitate the development of the GAMS files corresponding to the optimization problems that need to be solved at each step of the workflow, a template approach has been developed.
For each of the problem formulations included in the workflow (deterministic, under uncertainty, etc.) optimization template files have been developed in GAMS, both for the single- and multi-stream problem definition, and according to non-linear and piecewise linear capital cost constraint definition. These template files correspond to generic formulations of the optimization problems described in chapter 4 based on the generic process interval model structure presented in chapter 3, and are solved according to the direct solution method suitable to their problem type (LP, NLP, MILP, MINLP).
An overview of the template files is reported in table 7.1, where the name of the template file corresponding to each formulation is reported.

Table 7.1. Template optimization files developed in GAMS for the execution of the different steps of the workflow

#	problem	single-stream		multi-stream	
		non-linear capital cost	**piecewise lin. capital cost**	**non-linear capital cost**	**piecewise lin. capital cost**
3	**deterministic**	S_det.gms	S_P_det.gms	M_det.gms	M_P_det.gms
4	**unc. mapping**	S_map.gms	S_P_map.gms	M_map.gms	M_P_det.gms
5	**uncertainty**	S_unc.gms	S_P_unc.gms	M_unc.gms	M_P_det.gms
6	**flexible**	S_flex.gms	S_P_flex.gms	M_flex.gms	M_P_det.gms
7	**report**	S_rep.gms	S_P_rep.gms	M_rep.gms	M_P_det.gms

In order to allow their use for the formulation of different problems, the template files employ a data-independent formulation of the optimization problem, and the GDX functionality of GAMS (GAMS Development Corporation, 2011) is exploited for data input-output from and to binary and Excel files.
As a result, the dataflow related to the execution of each template file is organized as shown in figure 7.10: through GDX functionality, problem specification data are read from a given excel or binary file, together with the initial value for the optimization variables (if available). After the optimization problem is solved, the output GDX functionality is used to generate the result file.
The execution of the template files, therefore, requires the specification of the parameters required by the GDX functionality for the execution of the above described dataflow, which include the name and path of the input and output files, as well as data range and dimensions.
These optimization template files correspond to the most standard formulation of each of the described optimization problem, and are consequently suitable for simple standard problems, which can be solved through direct solution.
When the formulation of customized or complex problem is required, the optimization template files can constitute a good starting point for time-effective development of more sophisticated problem formulations.

A schematic representation of the execution of the entire workflow, based on the

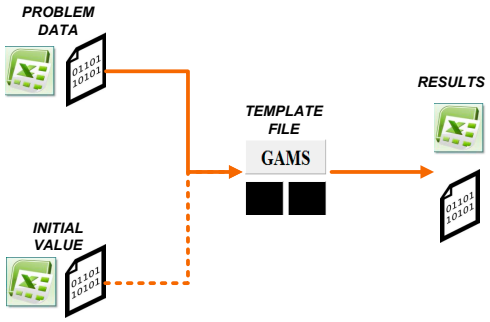


Figure 7.10. Schematic representation of the dataflow associated to a template file, where the dashed line represents an optional input

successive solution of different optimization template files is reported in figure 7.11, where the integration of dataflow and software tools is highlighted.

7.3 Implementation of bi-level solution methods

When standard problems (based on the generic interval model formulation) are concerned, a practical implementation of the bi-level decompositions algorithms presented in chapter 5.2 has been developed, allowing the automation of some of the steps required for their execution. This automation is realized through a development of 3 solution programs implemented in GAMS, for the execution of the 3 different solution methods.
Although differences between these solution files exist (reflecting the differences between the bi-level decompositions with respect to selection and relaxation of the complicating constraints), these programs are characterized by a common structure. Furthermore, these solution programs are generic, and therefore can be used for the solution of all problems generated according to the standard formulation.
A schematic representation of the workflow needed for the execution of the bi-level decomposition with the support of the described solution files is shown in figure 7.12.
As first step of the workflow, the user is requested to define the termination criteria and to specify the path where the GAMS file and the data file generated by EOLO are located.
Then, the user executes the solution file, which automatically performs a series of tasks (represented in blue), corresponding to one major iteration of the decomposition algorithm. In particular, the program loads the solution of previous iterations, based on which integer cuts are generated. The relaxed sub-problem is then formulated and solved, identifying an upper bound for the objective function value and a topology. The topology is fixed and the local problem is formulated and solved, resulting in a lower bound of the objective function. Finally, upper and lower bounds, topology and other relevant results are saved in a binary file, where the solution of each iteration is stored, and the automatic section of the solution is concluded.
At this point, the user analyzes the results; if any termination criterion is met, the

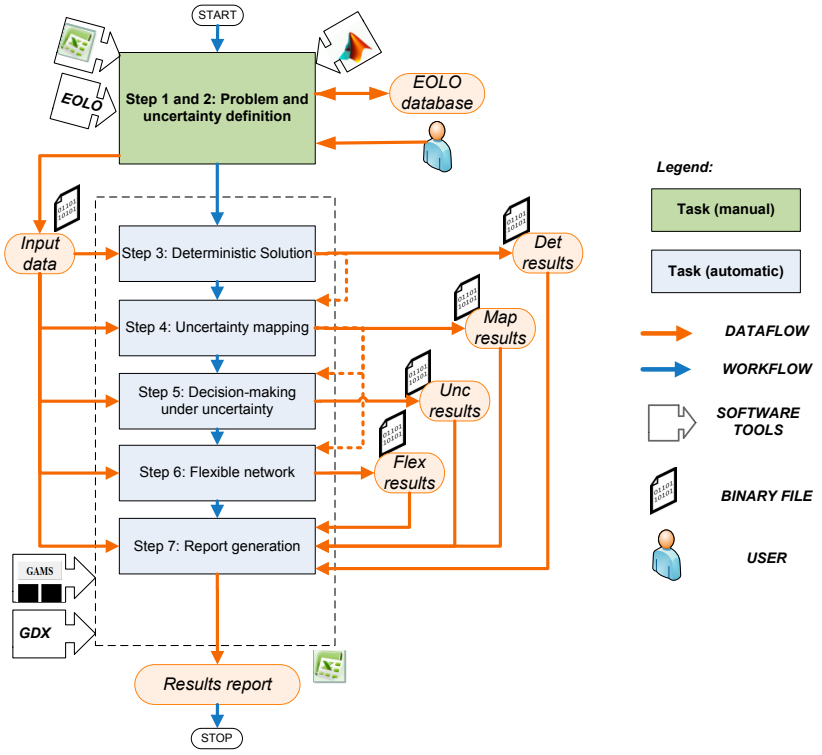


Figure 7.11. Schematic representation of workflow and dataflow in the practical implementation of the framework

solution is stopped, and the best local solution identified is reported. Otherwise, the user starts a new iteration, by executing the solution file again. The procedure is repeated until one of the termination criteria is met.
Even though the automation of the solution is not realized to its full extent, the described approach simplifies the execution of the bi-level decompositions, by automating some of the time consuming tasks such as problem reformulation, as well as facilitating the dataflow, with respect to storage of upper and lower bounds and generation of the integer cuts.

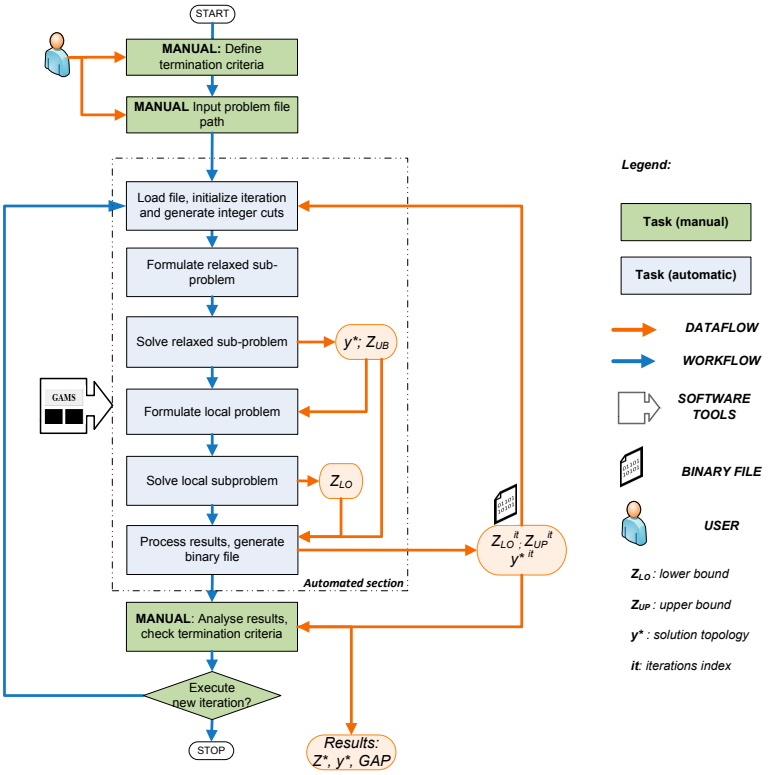


Figure 7.12. Schematic representation of the practical implementation of bi-level solution strategy. Upper and Lower bound are referred to maximization problem

Part III

Case Studies

Introduction to the case studies

In the third part of the thesis, the features of the integrated business and engineering framework for synthesis and design of processing networks are highlighted, through the formulation and solution of 3 case studies.
The first case study, called *Network Benchmark Problem* (NBP) is a numerical example, proposed in order to demonstrate the framework as well as test and benchmark the different solution methods presented in chapter 5.
The second case study, called *Soybean processing*, is a large synthesis and design problem of industrial relevance, developed in collaboration with the partner company Alfa Laval, in order to demonstrate the features of the framework with respect to handling large and complex problems, as well as the value of considering uncertainty in the decision-making process.
The third and last case study, called *Oil refinery wastewater treatment and reuse*, aims at exploring one of the most relevant applications of the framework, which is the design of wastewater treatment and reuse networks. Through this case study, the problem of integration of wastewater specific knowledge within the synthesis and design problem is explored, and the solution algorithm for the solution of large multi-stream problems is explored.
In the following chapters, these 3 case studies will be presented. The formulation and solution of the design problems will be briefly explained, in relation to the structure of the framework presented in the previous chapters.
For each case study, statistics related to the computational time required for the solution of the associated optimization problem are reported, as an indication of the computational resources needed. The CPU time reported in the following chapters are related to the solution performed on a standard computer, equipped with CPU Intel Core i5 2.53GHz.

8

Network Benchmark Problem

The Network Benchmark Problem (NBP) is a small numerical example that is proposed for demonstration and benchmarking of the framework. All NBP's data and models are reported in appendix C. In this chapter, the application of the framework to the solution of the NBP is described.
In order to demonstrate different features of the framework, the NBP will be solved for 2 cases. In the first case, called *NBP solution 1*, the single stream formulation of the NBP problem will be solved subject to data uncertainty, according to the workflow described in chapter 4. The obtained results will be analyzed, in order to highlight the information content and the value of the results report generated in the last step of the framework.
In the second case, (called *NBP solution 2*) a multi-stream formulation of the NBP problem will be solved in deterministic conditions, using different solution methods among the ones presented in chapter 5, in order to illustrate and benchmark them.

8.1 NBP solution 1 (single-stream)

8.1.1 Step 1: Problem formulation

The objective of NBP is the synthesis and design of the single-stream processing network that maximizes the Earnings Before Interest and Tax (EBIT), under uncertain conditions.
The superstructure is composed of 2 possible raw materials, 6 process intervals organized in 3 processing steps, and 4 potential products (see figure 8.1).
The component list contains 4 process components (C-1, C-2, C-3, C-4) and one utility component (U-1).

Data collection and model specification

In agreement with what stated in note 1 (section 5.2.1), the problem is formulated using piecewise linearization of the capital cost constraints, resulting in a MILP problem.
Despite being a small problem, the formulation of the NBP requires the definition of 2260 equations and the specification of 1299 data.
The formulation task is performed through the software EOLO, in order to test its features and evaluate the support that such a software tool can offer to the workflow of problem specification.
The first step of the formulation workflow is the specification of data related to alternatives, which constitute the first layer of the data architecture presented in

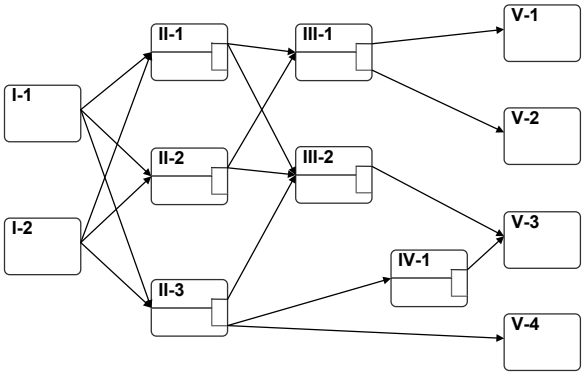


Figure 8.1. NBP superstructure

section 6.2. The list of process intervals, of process steps, of process and utility components and reactions is therefore defined, through the manual specification of 24 data (12 intervals, 5 steps, 5 components, 5 relative to the allocation of these components to the utility list and 2 reaction).
As a second step, components data such as molecular weight and utility price are defined. Since only one component has been identified as utility in the previous layer, the utility price for the remaining 4 components is automatically set to 0. As a consequence, out of the 10 data specified in this step, 4 are automatically fixed by the software based on a value relation between layer 1 and layer 2 data.
The third step is constituted by the specification of data related to intervals (third layer). Those include raw materials composition and price, products price, utility consumption, waste production and product separation data, as well as capital cost data. A total of 648 data are specified in this step, out of which 464 are automatically specified due to value of data set in previous layers. Moreover, 17 consistency checks are automatically imposed on the manually specified data, as described in section 6.2.
In the fourth step connectivity data are specified. Out of the 288 data fixed in this step, 174 are automatically specified, and 42 consistency checks are enforced.
The fifth step consists of the definition of data related to reactions (fifth layer). Stoichiometry and conversion for each of the reactions are defined; 38 data are specified by the user, 207 are automatically fixed and 8 consistency checks are imposed.
In the sixth step 72 data related to limits are specified, subject to 2 consistency checks.
Finally, in the last step of the problem formulation miscellaneous data are specified. This layer includes the time horizon for the calculation of the EBIT, the price for the emission of wastes, and the grid points for the piecewise linearization of the capital cost constraints. A total of 12 data is specified at this step.
The problem formulation statistics reported in table 8.1 show that all 1299 required data are specified through EOLO.
Moreover, 849 data (65% of the total) are automatically specified based on upper-

layer data, reducing the number of manually specified data to 440. Finally, 70 consistency checks are automatically defined and imposed on the manually specified data, contributing to guarantee consistent problem formulation.
Furthermore, EOLO automatically generates a GAMS readable file, which can be executed in order to obtain the deterministic solution (step 3 of the workflow), without requiring any additional programming.

Table 8.1. Statistics for NBP problem formulation

	n specified data	n manually specified data	n automatically specified data	n consistency checks imposed
Alternatives	24	24	-	1
Components	10	6	4	-
Intervals	648	184	464	17
Connectivity	288	114	174	42
Reactions	245	38	207	8
Limits	72	72	-	2
Miscellaneous	12	12	-	-
Total	1299	450	849	70

8.1.2 Step 2: Uncertainty domain definition

The second step of the workflow focuses on the characterization of the domain of uncertainty for the analysis.
In this case, 5 data are identified as sources of uncertainty: 2 raw material prices, 1 product price and 2 raw material compositions data.
The uncertainty associated to these data is described in terms of probability distribution, as reported in table 8.2.
The correlation between the uncertain data is described in terms of correlation between each data pair, which is reported in table 8.3. As an example, the correlation between the two raw material price data (π_R^{I-1} and π_R^{I-2}) is defined as 0.8, indicating the existence of a strong positive correlation between these data. The uncertain parameter space is sampled using Latin Hypercube Sampling (LHS) technique, and 200 future scenarios with equal probability of realization are generated, and reported in figure 8.2.
The correlation between the uncertain data is reflected in the sampling procedure by

Table 8.2. NBP solution 1: probability distribution of the uncertain data

Data	Mean	Probab. distr.	Min	Max	Description
π_R^{I-1}	16	Uniform	13.92	18.08	raw material I-1 price
π_R^{I-2}	18	Uniform	16.92	19.08	raw material I-2 price
π_P^{V-3}	70	Uniform	52.5	87.5	Product V-3 price
$\phi^{C-2,I-1}$	15	Uniform	10.5	19.5	comp. C-2 flow in raw material I-1
$\phi^{C-2,I-2}$	20	Uniform	8	32	comp. C-2 flow in raw material I-2

Table 8.3. NBP solution 1: correlation matrix for the uncertain data

	π_R^{I-1}	π_R^{I-2}	π_P^{V-3}	$\phi^{C-2,I-1}$	$\phi^{C-2,I-2}$
π_R^{I-1}	**1**	0.8	0.4	-0.2	0
π_R^{I-2}	0.8	**1**	0.4	0	0
π_P^{V-3}	0.4	0.4	**1**	0	0
$\phi^{C-2,I-1}$	-0.2	0	0	**1**	0.4
$\phi^{C-2,I-2}$	0	0	0	0.4	**1**

using the rank correlation method of Iman and Conover (1982), as shown in figure 8.2.

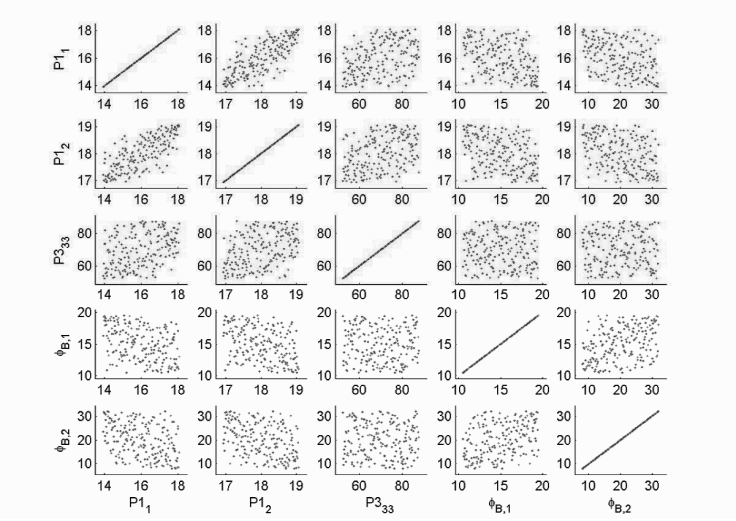


Figure 8.2. NBP solution 1: sampling

8.1.3 Step 3: Deterministic solution

The uncertain data are fixed at their expected values, and a deterministic MILP problem constituted by 2260 constraints and 120 binary variables (12 associated to the selection of process intervals and 108 related to the piecewise linearization of capital cost constraints) is formulated and solved in GAMS, using CPLEX solver version 12.
As a result the optimal network reported in figure 8.3 is identified, and a corresponding EBIT value of 4038.826 $k\$/yr$ is calculated.
Stream table and utility table for the results are shown in tables 8.4 - 8.5

Table 8.4. NBP solution 1: deterministic solution - stream table

	II-2	**III-1**	**V-1**	**V-2**
C-1	78.0	78.0	54.6	23.4
C-2	15.0	0.8		0.8
C-3	7.0	19.5		15.6
C-4		12.5		12.5

Table 8.5. NBP solution 1: deterministic solution - utility table

	II-2	**III-1**
uA	22.5	70.2

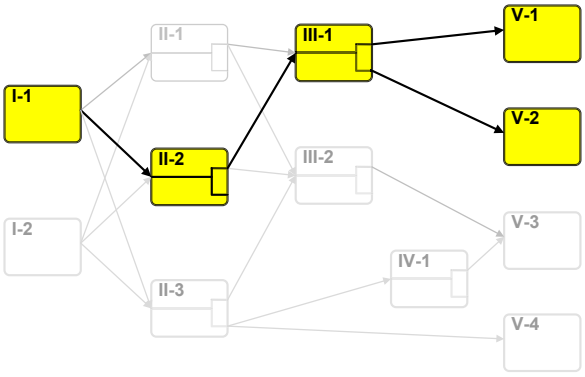


Figure 8.3. NBP solution 1: deterministic solution

8.1.4 Step 4: Uncertainty mapping

For each of the 200 future scenarios generated in step 2, a separate optimization problem is formulated and solved.
The results are summarized in figure 8.4. On the left hand-side of the figure, the cumulative probability distribution of objective function value is reported. The interpretation of the plot is the following: for every value of EBIT represented on the x-axis, the probability of obtaining an objective function value greater or equal is read on the y-axis. For this case, a large variability of objective function value is observed for different future scenarios, with EBIT values spanning from 2399.5 to 5846.9 $k\$/yr$.
On the right hand-side, the different topologies that have been identified as optimal for different realizations of the uncertain data are plotted against their probability. For this case, the exploration of the uncertainty domain defined in step 2 leads to the identification of 6 different networks.
More details with respect to the different topologies obtained are reported in tables 8.6 and 8.7. From these tables, it can be seen that none of the intervals is always part of an optimal network, while 2 of them (II-3 and IV-1) are never part of one.
These results indicate that the uncertainty in the data has a large impact on the decision-making and on the performances metrics, and therefore needs to be carefully considered in the decision-making process.

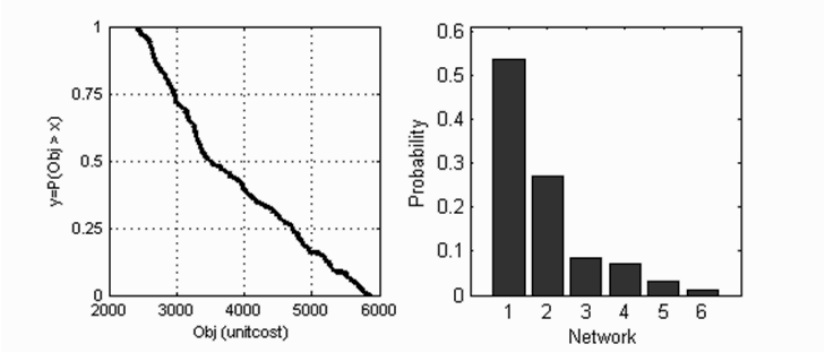
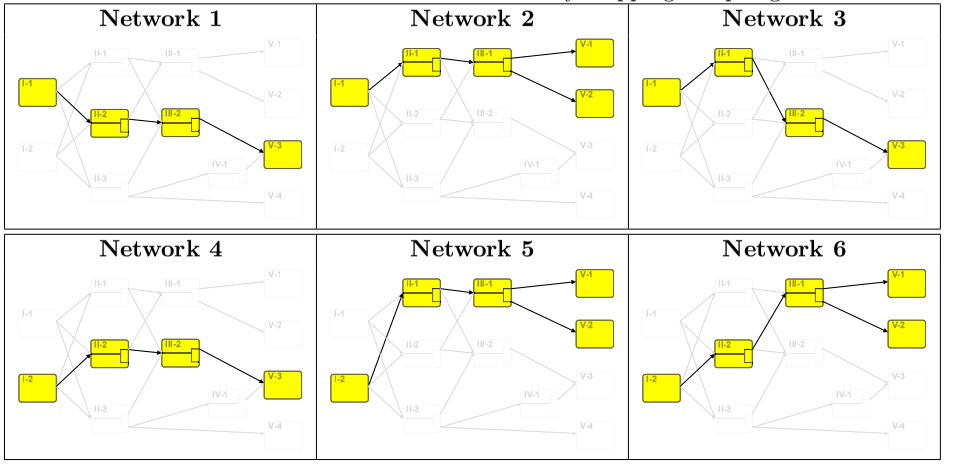


Figure 8.4. NBP solution 1: Uncertainty mapping - Results

Table 8.6. NBP solution 1: Uncertainty mapping - Results details

Network	Probability	Selected Intervals											
		I-1	I-2	II-1	II-2	II-3	III-1	III-2	IV-1	V-1	V-2	V-3	V-4
1	53.50%	1	0	0	1	0	0	1	0	0	0	1	0
2	27.00%	1	0	1	0	0	1	0	0	1	1	0	0
3	8.50%	1	0	1	0	0	0	1	0	0	0	1	0
4	7.00%	0	1	0	1	0	0	1	0	0	0	1	0
5	3.00%	0	1	1	0	0	1	0	0	1	1	0	0
6	1.00%	0	1	0	1	0	1	0	0	1	1	0	0

Table 8.7. NBP solution 1: Uncertainty mapping - Topologies



8.1.5 Step 5: Solution under uncertainty

The problem under uncertainty is formulated and solved using sample average approximation (SAA), resulting in the formulation of a deterministic MILP equivalent constituted by 398,669 constraints and 120 binary variables.
The optimal network under uncertainty, reported in figure 8.5, is different from the optimal solution obtained under deterministic conditions, and a different interval is selected for step 2. The expected value of EBIT is calculated as 2778.089 $k\$/yr$.

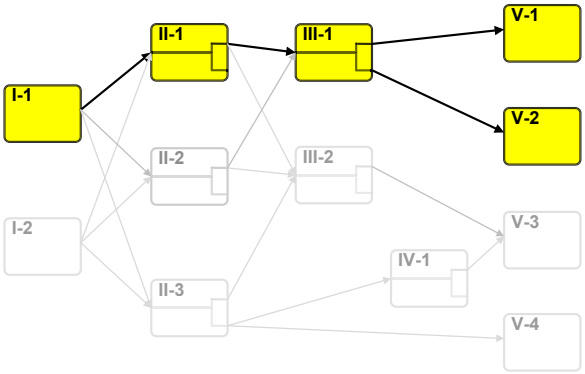


Figure 8.5. NBP solution 1: solution under uncertainty

8.1.6 Step 6: Optimal flexible solution

The flexible problem is formulated as a two stage stochastic programming, as described in section 4.6, resulting in the formulation of a deterministic equivalent MILP problem constituted by 609,910 constraints and 2508 binary variables, which is solved via the bi-level decomposition approach presented in section 5.2.2.
The solution is obtained in 3 iterations and the solver stops because of deterioration of the upper bound. The progress of the lower and upper bound over the proceeding of the algorithm are reported in table 8.8.
As a result, the network reported in figure 8.6 is identified. For the first 2 steps, 2 intervals are selected as part of the optimal network, resulting in a redundant structure. Such a redundancy allows a greater operational flexibility, which can be exploited at operational stage once additional knowledge on the uncertain data becomes available, to select the most profitable operational policy.
The increase in operational revenues which is obtained, compensates the additional capital investment required, resulting in an expected EBIT of 3297.947 $k\$/yr$, corresponding to an increase of more than 5.5% with respect to the optimal network under uncertainty. Such an improvement is obtained thanks to the integration of operational flexibility considerations in the design decision-making phase.
The solution statistics with respect to the NBP problem under uncertainty in the different steps of the workflow are reported in table 8.9.

Table 8.8. NBP solution 1: flexible solution through bi-level decomposition

Iteration	Bounds	Topology
iteration 1	UB_1=3736.408 LB_1=2288.271	I-1; I-2; II-1; II-2; III-1; III-2; V-1; V-2; V-3]
iteration 2	UB_2=3508.548 LB_2=**3099.514**	I-1; I-2; II-1; II-2; III-2; V-3
iteration 3	UB_3=2891.587 LB_3=2068.667	I-1; I-2; II-1; II-2; III-1; III-2; V-1; V-2

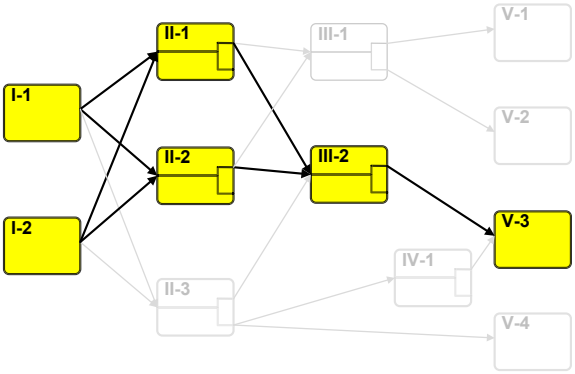


Figure 8.6. NBP solution 1: flexible solution

Table 8.9. NBP solution 1: statistics

Problem	Deterministic	Under uncertainty	Flexible
n binary variables	120	120	2508
n constraints	2,260	398,669	609,910
Relative optimality tolerance	1.00E-06	1.00E-06	1.00E-06
CPU time (s)	0.33	56.12	316.41
Solution algorithm	direct	direct	bilevel decomp.
n outer iterations	-	-	4
Average CPU time for 1 iteration (s)	-	-	79.1

8.1.7 Step 7: Report generation

As a final step, a report containing the above described results and the indicators is generated, providing an overview of all the information obtained and a documentation of the problem specification for which the results have been obtained. The summary section of the report, containing the most relevant information and results to be considered in the decision-making, is reported in table 8.10.
As the report shows, different networks are obtained as solution of the deterministic and stochastic problems.
Useful information with respect to the impact of data uncertainty on the decision-making problem and on the profitability of the investment are obtained through the calculation of the below mentioned indicators.
In particular, the Expected Value of Perfect Information (EVPI, equation 4.26) is $958.319 k\$/yr$ (24% of the EBIT). This value represents the expected performance gain, which could be obtained if the exact value of the uncertain data was known prior to taking the design decisions. As discussed in chapter 4, this indicator provides also a measurement of project maturity, and can be used to support process management decisions (along with other consideration, such as time-to-market, company strategy etc.).
An important indicator of the value associated with the execution of the design under uncertainty is constituted by the Value of the Stochastic Solution (VSS, equation 4.27). For the NBP problem, large VSS are obtained, indicating the importance of considering the uncertainty in the decision-making process. In particular, by allowing the identification of the flexible network structure reported obtained as solution of step 6, the framework is estimated to contribute to a return of 954.955 $k\$/yr$ in terms of EBIT.
The Uncertainty Price (UP, equation 4.28) shows the decay in financial performances of the investment, due to the need of considering uncertainty in the decision-making. The effectiveness of the flexible network in making use of network flexibility to mitigate the consequences of the uncertainty is shown by the reduction in UP (-38%) with respect to the optimal network under uncertainty.
On the base of these results, the analysis suggests the flexible network as the best decision.

Table 8.10. NBP solution 1: Solution Report

Solution	**Network**	**EBIT (k$/yr)**
Deterministic network	I-1; II-2; III-1; V-1; V-2	4038.826
Network under uncertainty	I-1; II-1; III-1; V-1; V-2	2778.089
Flexible network	I-1; I-2; II-1; II-2; III-2; V-3	3099.514

Indicator	**VSS (k$/yr)**	**UP (k$/yr)**	**EVPI (k$/yr)**
Network u/ uncertainty	633.529	1260.737	958.319
Flexible network	954.955	939.312	

8.2 NBP solution 2 (multi-stream)

8.2.1 step 1: Problem formulation

The formulation of the NBP solution 2 is based on the modification of solution 1, described in section 8.1.1. With respect to the NBP solution 1, a limit to the maximum productivity of product V-1 is enforced, through the definition of constraint 8.1:

$$\sum_i \left(F_{IN}^{i,V-1} \right) \leq 50 \tag{8.1}$$

For the sake of focusing on the solution of multi-stream problems, uncertainty is not considered in the formulation NBP solution 2. Therefore step 2 and 4-7 of the workflow are not executed.

8.2.2 step 3: Deterministic solution

The multi-stream problem results in the formulation of a non-convex MINLP, with 2,260 equations and 120 binary variables.
In order to benchmark the performances of different algorithms, the problem has been solved according to different solution strategies, such as:

1. DICOPT
2. BARON
3. Bi-level decomposition algorithm
4. Multi-level domain partitioning

DICOPT

The first solution has been obtained via the OA/ER algorithm implemented in the solver DICOPT. While being able to solve extremely large problems, such a solver is not designed to handle non-convex constraints, since part of the feasible space is eliminated by the linearizations performed at each iteration. Consequently, its use for the solution of non-convex problems may lead to local optima or infeasibile solutions (Grossmann *et al.*, 2011).
In the case of the NBP solution 2, DICOPT identifies a local optimum, with a value of EBIT of 2204.14 $k\$/yr$.

BARON

As described in chapter 5, BARON is a global solver specifically designed to identify the global solution of non-convex problems. When applied to the NBP, BARON requires a CPU time of 7.48 seconds to identify a solution characterized by an EBIT value of 2318.09 $k\$/yr$. Moreover, BARON is able to prove the global optimality of the obtained solution through the convergence of the bounds.
The results (reported in figure 8.7 and table 8.11) show that, in order to satisfy the limit imposed by equation 8.1 on the maximum productivity of V-1, the outlet flow

of interval II-1 is split and 91% of the flow is fed to III-1, while the remaining 9% is allocated to interval III-2, resulting in the production of product V-3.

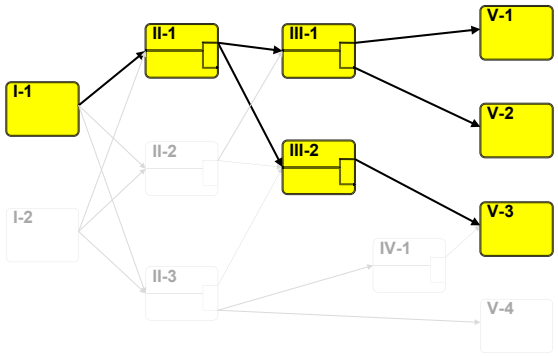


Figure 8.7. NBP solution 2 (multi-stream): topology of the global optimum solution

Table 8.11. NBP solution 2 (multi-stream): Stream table of the global optimum solution

$F_{IN}^{i,k}$	**II-1**	**III-1**	**III-2**	**V-1**	**V-2**	**V-3**
C-1	78.00	71.23	6.77	49.86	21.37	6.77
C-2	15.00	1.37	0.13	0.14	1.23	0.08
C-3	7.00	6.39	0.61		5.11	0.06
C-4						0.55

Bi-level decomposition algorithm

The third solution method consists in the execution of the bi-level decomposition algorithm for multi-stream problems, which have been presented in section 5.2.2, supported by the described domain partitioning and strengthening cuts strategies. For the domain partitioning, a 2x2 equidistant grid is implemented.
The relaxed problem identifies an upper bound of 2419.37 $k\$/yr$, and the lower bound of 2318.09 $k\$/yr$ (corresponding to global optimum) is identified in the first iteration, as solution of the local sub-problem.
The procedure is stopped for deterioration of the upper bound; a CPU time of 1.33 seconds is required for the identification of the global optimum.

Multi-level decomposition algorithm

The fourth method applied to the solution of the NBP is the multi-level algorithm, which has been described in section 5.2.2 and figure 5.4. The grid for the first level relaxation is constituted by an equally spaced 2x2 grid, equal to the one employed in the previous case. The second level grid is built partitioning each of the active intervals through a 2x2 grid.
The first level relaxation corresponds to the upper bound sub-problem solved with the previous method, and leads to the identification of the same upper bound. In

the second level, the resolution of the grid is increased in the vicinity of the solution identified in the first level, resulting in a tighter relaxation. As a consequence, the value of the upper bound is improved to 2336.20 $k\$/yr$. The local problem leads to the global optimum solution of 2318.09 $k\$/yr$ identified earlier.
The procedure is stopped because of deterioration of the upper bound. In total, a CPU time of 2.69 seconds is required or the identification of the global optimum, with a relative optimality tolerance of 0.8%.
In order to study the influence of the definition of the domain partitioning grid on the performances of multi-level decomposition algorithm, the problem has been solved for different grid definitions. The results, summarized in table 8.13, show that the relaxed sub-problem was able to identify the correct topology at the first iteration in all cases. The grid definition, as expected, appears to have a large impact on the relative tolerance obtained, as well as on the computational time required for the solution.

8.2.3 Comparison of the solution methods

From the comparison of these 4 solution methods (summarized in table 8.12), a number of considerations can be made. As expected, DICOPT has not been able to identify the global optimum of the non-convex problem, and returned a local solution whose performances (in terms of objective function value) were 5% worse.
On the contrary, BARON has shown the capability of solving a problem of this size and complexity to global optimality, as well as to prove the optimality through the convergence of the two bounds. The problem has been solved via direct solution, without requiring any problem reformulation or conditioning. Because of its ease of use and flexibility, therefore, BARON appears as the most suitable option for the solution of small problems.
Both bi-level and multi-level decomposition have shown the ability of solving the problem to global optimality, in a fast and reliable manner. In particular, the multi-level relaxation has proven the ability of tightening the upper bound (for maximization) to a considerable extent, while requiring moderate extra computational resources. Because of this feature, the multi-level relaxation appears to be a suitable method for the solution of large multi-stream problem.
As described in section 5.2, from a user perspective the main drawback associated to the use of bi- and multi-level decompositions lays in the problem reformulation that these strategies require.
This observation underlines the significance of tools integration, which constitutes one of the key features of the developed framework. The integration of models and solution strategies within a common framework, in fact, enables the automation of the required reformulations (as described in section 7.3), contributing to manage the complexity associated to these algorithms.

Table 8.12. NBP solution 2: comparison of solution methods

Solver	LB	UB	Relative Tolerance	Gap from global optimum	CPU time (s)	Solution Topology
Dicopt	2202.14	2202.14	1E-06	5.0%	1.12	I-1, II-1, III-2, V-3
Baron	**2318.09**	2318.09	1E-06	-	7.38	I-1, II-1, III-1, III-2, V-1, V-2, V-3
bi-level	**2318.09**	2419.37	4.4%	-	1.33*	I-1, II-1, III-1, III-2, V-1, V-2, V-3
multi-level	**2318.09**	2336.20	0.8%	-	2.69**	I-1, II-1, III-1, III-2, V-1, V-2, V-3

* sum of the CPU time required for the solution of the UP problem and of the LB problem.
** sum of the CPU time required for the solution of the two UP problems and of the LB problem.

Table 8.13. NBP solution 2: comparision of partitioning strategies in multi-level domain partitioning

N_o	N_p	$Nref$	UB	Relative Tolerance	CPU time (s)	UB Topology
1	2	2	2394.46	3.3%	0.56	I-1, II-1, III-1, III-2, V-1, V-2, V-3
2	1	2	2339.94	0.9%	0.72	I-1, II-1, III-1, III-2, V-1, V-2, V-3
2	2	2	2336.20	0.8%	1.69	I-1, II-1, III-1, III-2, V-1, V-2, V-3
2	2	3	2343.00	1.1%	3.14	I-1, II-1, III-1, III-2, V-1, V-2, V-3
2	2	4	2329.77	0.5%	7.46	I-1, II-1, III-1, III-2, V-1, V-2, V-3

where N_o and N_p are the number of first-level partitions for variable $Fout$ and SM respectively; $Nref$ is the number of second level partition of the active interval.

8.3 Conclusions from the NBP case study

In this case study, the NBP has been used as a numerical example in order to demonstrate the features of some of the key elements integrated in the framework. In particular, through the NBP solution 1 the simplification of the data specification workflow obtained by the development of the software EOLO has been shown.
Furthermore, the workflow for synthesis and design of a processing network under uncertainty has been demonstrated, highlighting the increase in mathematical complexity that derives from the inclusion of uncertainty (with respect to number of constraints and variables) in the decision-making process.
The capability of the framework to manage such a complexity has been highlighted by solving the problem under uncertainty, obtaining a robust and a flexible solution. Finally, the informative value of aggregating all the results obtained in a report, containing the most relevant outcomes as well as some indicators useful to understand the impact of uncertainty on the decision-making problem, has been demonstrated.
With the solution of the NBP solution 2, moreover, the capability of the framework to deal with the additional complexity of multi-stream cases, requiring the formulation of non-convex problems, has been demonstrated.
In particular, the comparison between the solution strategies has indicated that multi-level decomposition may lead to the identification of good upper bounds for the objective function, at a moderate computational cost. Consequently, this strategy appears as a suitable method for the solution of large non-convex problems, which cannot be solved directly through BARON due to their size and complexity (Sahinidis and Tawarmalani, 2011).

9

Soybean Processing

The second case study is related to the problem of synthesis and design of a processing network for soybean processing.
Soybean (*Glycine max*) has become one of the most important agricultural commodities, with a steadily increasing global production, which reached 248 MMT in 2009 (Thoenes, 2009). The average composition of soybean is reported in table 9.1 (Hammond *et al.*, 2006).

One of the reasons motivating its importance in the agri-food business lays in the variety of its use. Soybean, in fact, can be used as a raw material for a wide range of food, feed and pharma products: soybean oil, for example, is widely used as cooking or dressing oil, but can also have feed or technical applications, or be used as raw material for biodiesel production. Furthermore, defatted soy-beans are a low-cost source of protein, used to substitute animal protein in a wide range of feed and food products. Moreover, several soy by-products have specific functionalities, which are exploited in a number of high-end applications. As an example, soy lecithin is used in a wide variety of food and pharmaceutical products as emulsion stabilizer. Tocopherols finds application as antioxidant in pharmaceuticals, as well as natural preservatives for packaged food. Other by-products such as hulls and fatty acids are mostly used as feed ingredients.
Soybean processing being a low margin operation, profitability is achieved only if all different seed components are allocated to commercially valuable products and by-products, in an optimal manner (Chicago Board of Trade, 1998). Determining the optimal allocation of the different components of this resource is a relatively complex task, because of the wide spectrum of potential products, as well as of the mutual interactions between them.
Since the market of agricultural commodities is highly volatile, the optimal resource

Table 9.1. US average soybean composition, dry basis (Hammond *et al.*, 2006)

Component	% **weight**	**Standard deviation**
Protein	40.69%	0.51
Lysine	2.56%	0.11
Methionine	0.57%	0.03
Cysteine	0.72%	0.06
Tryptophane	0.52%	0.05
Threonine	1.54%	0.07
Oil	21.38%	0.64
Ash	4.56%	0.34
Carbohydrate	29.4%	3.29

allocation changes frequently over time, urging soybean processing companies to rethink their product portfolio frequently (Thoenes, 2009).
Although market-related models exist, to support the decision-making related to the trading of such resources (Chicago Board of Trade, 1998), to the best of our knowledge we are not aware of the existence of any decision-making tool to support synthesis and design of optimal processing networks in particular by realizing the integration of the market- and engineering-related dimensions of the problem. Consequently we argue that the soybean industry sector in particular (and in general all the agro-industrials sector), could benefit from the development and the application of the integrated business and engineering framework, because of the reasons discussed in section 1.1. For this reason, the problem of soybean processing has been selected as a relevant case study.
The case study has been developed in collaboration with Alfa Laval, a global and well established player in the oilseed processing business. The partnership with Alfa Laval facilitated the gathering of all data and models required for the problem formulation. Moreover, it provided a mean of validation of the results obtained from the analysis. To protect the confidentiality of Alfa Laval's proprietary data, all results reported in this chapter are based on scaled measurement units called unit-mass and unit-cost, equivalent to mass and cost units.
In order to highlight the versatility of the tool, the problem is solved for 2 cases, denominated *Soybean processing solution 1* and *Soybean processing solution 2*. In *Soybean processing solution 1*, the design problem is solved for deterministic conditions, with the goal of maximizing the Gross Operating Income (GOI). Four different scenarios are considered with respect to data values, and the differences in terms of the obtained results are discussed. In *solution 2*, the problem of processing network design under uncertainty is considered, aiming at the maximization of the expected Earning Before Interests and Tax (EBIT), in an environment characterized by market, technical and supply uncertainty.

9.1 Soybean processing solution 1

9.1.1 Step 1: Problem Formulation

Problem Definition

The problem is defined as a resources allocation problem, assuming the availability of 100 unitmass of 2 soybean supplies, characterized by different quality (with respect to their lecithin content). The design of a new processing network is studied, and therefore no topological constraints related to pre-existing processing capacity are considered. Gross Operating Income (GOI) is selected as the objective function (equation 3.35). For the sake of simplicity, production of biodiesel is considered out of the scope of the problem.
All knowledge and data relevant to the problem are collected by integrating the information available in the open literature with the industrial knowledge of Alfa Laval.

Superstructure definition

The superstructure generation method described in section 6.1 (figure 6.1) is used in order to define the search space for the optimization problem. In the next section, the application of the method to the soybean case study is described.

Superstructure Generation Method

Step 1: Problem definition The goal is defined as the synthesis of a superstructure for soybean processing, employing standard edible oil processing technologies. The scope is to cover all the value chain of soybean-related products, including food, feed and pharmaceutical ingredients production. As stated above, biodiesel production is not included in the scope.

Step 2: Raw materials properties, products properties and synthesis rules definition As stated in the introduction, the raw material list contains soybean of 2 different qualities with respect to lecithin content (reported in table 9.2, where lecithin is identified as Hydratable Phosphorous, HP). Based on information in the open literature and on Alfa Laval expertise, a product list constituted by 16 potential products, covering a wide spectrum of soy-based products such as oil, meal and pharmaceutical ingredients is compiled, as reported in table 9.3.
Finally, the synthesis rules reported in table 9.4 are generated, based on commercial, engineering and regulatory insights from Alfa Laval or from the open literature.

Table 9.2. Raw Material list

	R1	R2
Components*	Low Quality Soybean	High Quality Soybean
TAG	18.10%	19.00%
DAG	0.20%	0.10%
MAG	0.10%	0.00%
FFA	0.10%	0.07%
HP	0.20%	0.40%
NHP	0.60%	0.40%
water	12.50%	12.50%
tocopherols	0.04%	0.04%
tocotrienols	0.01%	0.01%
sterols	0.07%	0.07%
protein	36.00%	38.00%
fiber	17.00%	14.40%

*TAG: triacylglycerides, DAG: diacilglicerides, MAG: monoacilglicerides, FFA: free fatty acid, HP: hydratable phosphorous (lecithin), NHP: non-hydratable phosphorous

As stated in note 4, the execution of the superstructure generation method becomes complex and cumbersome, when such a large number of options are considered for raw materials and products. In order to cope with this complexity, the incremental superstructure synthesis procedure described in section 6.1 is adopted for this case study, and the synthesis procedure is executed iteratively, for different definitions of raw materials and products groups.
In the next paragraphs, the execution of the iterative procedure is described.

Table 9.3. Product list

Tag	Name	Composition	Process requirements
P1	Pressed Oil	Oil	No solvent extraction
P2	Refined Oil	Oil	
P3	Tocopherols	Tocopherol	
P4	Tocotrienols	tocotrienol	
P5	Toco Mix	Tocopherol, tocotrienol	
P6	FFA	FFA	
P7	Sterols	Sterols	
P8	FADD	Tocopherol, tocotrienol, sterols, FFAA	
P9	Lechitin	Hydratable phosphorous 50%, oil 50%	
P10	Gums	Hydratable phosphorous 33%, oil 33%, water 33%	
P11	LowPro Meal	Protein 33%, Fiber 53% Oil 2%, Water 12%, Hexane¡150ppm	
P12	Enhanced Meal	Protein 33%, Fiber 53%, Oil 2% , Water 12%,Hexane¡150ppm	Enzymatic treatment for digestibility
P13	HighPro Meal	Protein 37%, Fiber 49% Oil 2%, Water 12%, Hexane¡150ppm	
P14	Protein Concentrate	Protein 65%, Fiber 35%	
P15	Fibers	Fiber 80%, Protein	
P16	Hulls	Fiber 80%, Oil, Protein Water	

Table 9.4. Synthesis rules defined based on Alfa Laval expertise

Rule	Description
R-1	Refining of lipids can be performed only after removal of proteins and fibers
R-2	Lipids/fibers separation cannot be performed after protein separation
R-3	Refining of oil microcomponents can be done only after microcomponents have been separated from the oil
R-4	Refining of phospholipids can be done only after Phospholipids separation from oil
R-5	The network should always lead to the production of oil (P1 or P2)
R-6	Mixing of previously separated compounds is not allowed
R-7	Arrangements which facilitate heat integration should be privileged
R-8	Meal treatment can be executed only after lipids separation

Step 2b (*iteration 1*): Raw material and product groups definition In this step, required for the execution of the incremental procedure, raw materials and products are grouped according to their properties.
As described, each iteration is based on a different definition of these groups. For the first iteration, a small number of coarse groups is defined; at each iteration, progressively finer groups are defined, until the overall complexity of the problem is considered. For this case study, 3 iterations of the incremental superstructure synthesis procedure are performed. The first 2 iterations are executed considering the raw materials and product groups reported in table 9.5, while the final iteration considers the original raw material and product list defined in step 2.
The interpretation of table 9.5 is the following: in the first iteration of the method, Tocopherols and Lecithin (which were identified as 2 separate products in table 9.3) are considered together as part of a *lipids* product group. In the second iteration, on the other hand, these products are considered in separate groups, as *microcomponents* and *phospholipids* respectively.

Table 9.5. Definition of raw material and product groups for the 2 iterations

RAW MATERIALS		**Grouping**	
Tag	**Raw Materials**	**iteration 1**	**iteration 2**
R1	Low-quality soybean	SOYBEAN	SOYBEAN
R2	High-quality soybean		
PRODUCTS		**Grouping**	
Tag	**Products**	**iteration 1**	**iteration 2**
P1	Pressed Oil	LIPID	OIL
P2	Refined Oil		
P3	Tocopherols		MICROCOMP.
P4	Tocotrienols		
P5	Toco Mix		
P6	FFA		
P7	Sterols		
P8	FADD		
P9	Lecithin		PHOSPHOLIPIDS
P10	Gums		
P11	LowPro Meal	PROT/FIBER	PROT/FIBER
P12	Enhanced LowPro Meal		
P13	HighPro Meal	PROTEIN	PROTEIN
P14	Protein Concentrate		
P15	Fibers	FIBER	FIBER
P16	Hulls		

For the first iteration, 1 raw material (soybean) and 4 product groups (lipids, protein, fibers and protein/fiber mixture) are considered.

Step 3 (*iteration 1*): Select Property difference of highest hierarchy The property difference occupying the highest hierarchical position is composition, therefore mixing and separation are employed to resolve it.

Step 4 (*iteration 1*): Steps superstructure generation The concentration difference between raw material and products is resolved through separation process steps. The separation of lipid, protein and fiber components are added to the superstructure, considering all possible combinations, resulting in a superstructure containing 4 configurations, as shown in figure 9.1.

Step 5 (*iteration 1*): Steps superstructure screening In this step, the configurations generated in the previous step are screened against the synthesis rules defined in table 9.4.
One of the configurations generated in the previous step is the separation of proteins first and then the separation of fat and fibers. According to rule R-2, lipids cannot be separated from fibers after proteins separation; therefore the separation of lipids and fibers realized in the second step is non-legal. This means that if proteins are separated first, no oil product can be produced, and rule R-5 is violated. Therefore, this configuration is eliminated by the superstructure, reducing the number of possible configuration from 4 to 3. All other configurations satisfy the synthesis rules, and are therefore considered legal and kept in the superstructure.
All property differences have been addressed, and the method continues with a new major iteration, through the definition of new raw material and product groups in step 2b. The superstructure obtained after the first iteration is shown in figure 9.1, where the eliminated configuration is reported in grey.

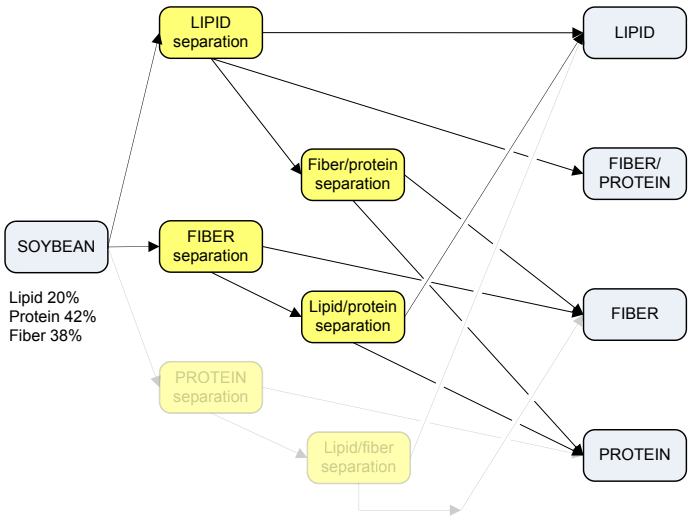


Figure 9.1. Steps Superstructure (iteration 1)

Step 2b (*iteration 2*): Raw materials and products group definition One raw material and 6 product groups are considered in the second iteration, as reported in table 9.5.

Step 3 (*iteration 2*): Select Property difference of highest hierarchy The property difference occupying the highest hierarchical position is composition, therefore mixing and separation are employed.

Step 4 (*iteration 2*): Steps superstructure generation The concentration difference between raw material and products is resolved through separation. Separation of oil, phospholipids and microcomponents are added to the superstructure. As described in chapter 4, the superstructure is built starting from the one obtained in the previous iteration. Consequently, configurations rejected in previous iterations are automatically excluded from the superstructure generation in the following iterations, allowing to reduce the number of infeasible configurations obtained at each step. As a result, a superstructure containing 72 configurations is obtained.

Step 5 (*iteration 2*): Steps superstructure screening The superstructure generated in the previous step is screened against the defined synthesis rules.
A large number of options is discarded, due to violation of rule R-1, which states that the separation of lipids from phospholipids and microcomponents can be carried out only after protein and fiber have been removed. As a result, only 9 legal configurations are kept into the superstructure.
All property differences have been addressed, and the method continues with a new major iteration, through the definition of new raw material and product groups in step 2b. The superstructure obtained after the second iteration is reported in figure 9.2.

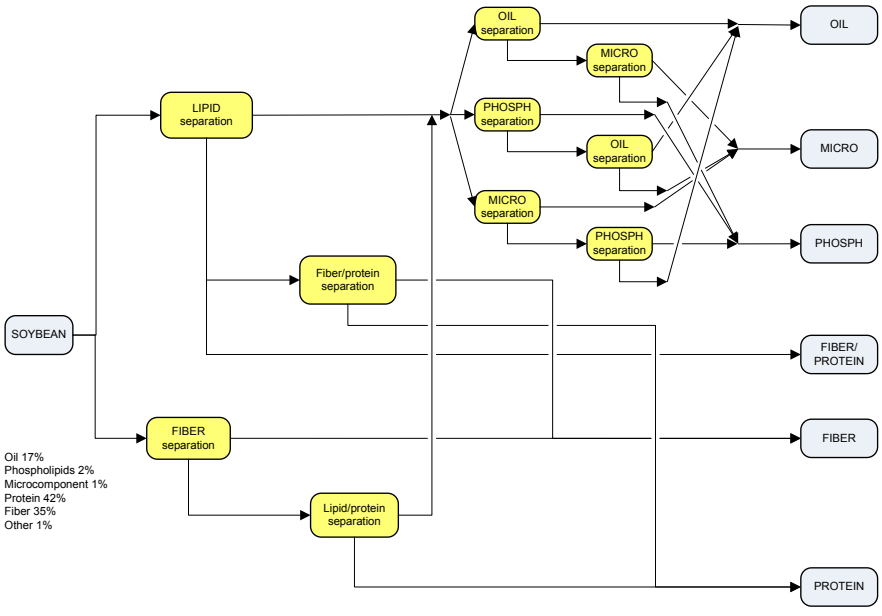


Figure 9.2. Steps Superstructure (iteration 2)

Step 2b (*iteration 3*): Raw materials and products group definition For the last iteration, the original list of raw materials and products is considered.

Step 3 (*iteration 3*): Select Property difference of highest hierarchy The property difference occupying the highest hierarchical position is composition. Moreover, the production of product P12 (low-protein meal with enhanced digestibility), requires an enzymatic treatment step in order to obtain the required functionality, therefore a meal treatment step is considered to address such a requirement.

Step 4 (*iteration 3*): Steps superstructure generation The differentiation of each product group in the individual products is executed through refining steps, which are added to the superstructure obtained in the third iteration, in all possible configurations. Similarly, the meal treatment enhancing digestibility is also included in the superstructure in all possible positions.

Step 5 (*iteration 3*): Steps superstructure screening The superstructure generated in the previous step is screened against the synthesis rules.
According to rules R-3 and R-4, microcomponents and phospholipids refining have to be executed last (after their separation from oil). Similarly, the meal treatment enhancing digestibility has to be placed as the last treatment, according to rule R-8. As a result of this screening, the steps superstructure reported in figure 9.3 is obtained.

Step 6: Process technologies identification and synthesis rules definition The knowledge base is searched, to identify process technologies which can execute each of the process steps contained in the superstructure. Each process interval is analyzed, in order to identify and define its utility requirements, as well as special wastes and byproducts, which are considered as secondary flows in the design phase. The list of process interval alternatives for each process step is reported in table 9.6.

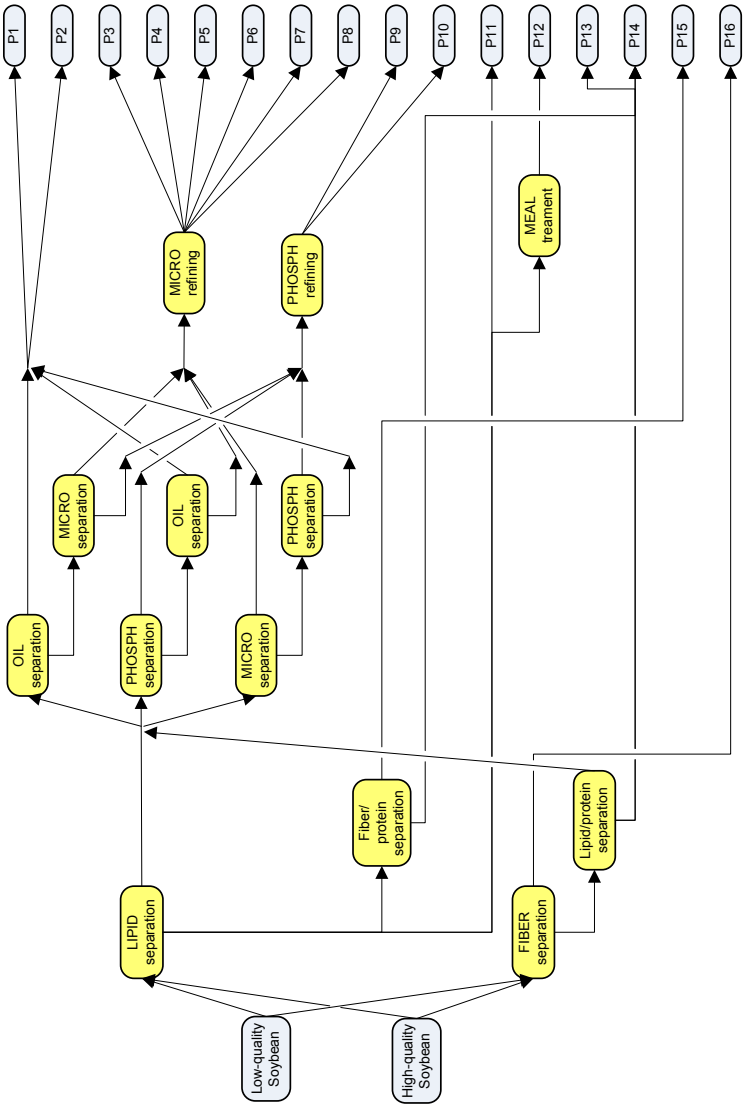


Figure 9.3. Steps Superstructure (final)

Table 9.6. Identification of process technologies for soybean processing

Process step	***#***	***Process interval***	***Notes***
Fiber separ.	4	Dehulling	
Lipids separation	5	Expelling	
	8	Hexane Extraction	Needs hexane, explosive, excess residual water and hexane in meal
	9	Hexane Extraction in heat integrated plant	Needs hexane, explosive, excess residual water and hexane in meal
	31	Low-pro meal dryer cooler	*(added to treat residual water and hexane in meal)*
	32	High-pro meal dryer cooler	*(added to treat residual water and hexane in meal)*
Phospholipids separation	11	Water degumming	
	12	Acid degumming	Needs phosphoric acid
	13	Enzymatic degumming	Needs enzymes
	14	Caustic degumming	Produces soaps
	16	Washing physical refining	
	18	Washing chemical refining	
Microcomp. separation	19	1 step bleaching (phys. ref.)	Produces spent clays
	20	2 step bleaching (phys. ref.)	Produces spent clays
	21	1 step bleaching (chem. ref.)	Produces spent clays
	22	2 step bleaching (chem. ref.)	Produces spent clays
	23	Deodorization (phys. ref.) design 1	Causes degradation of phospholipids
	24	Deodorization (phys. ref.) design 2	Causes degradation of phospholipids
	25	Deodorization (phys. ref.) design 3	Causes degradation of phospholipids
	26	Deodorization (chem. ref.) design 1	Causes degradation of phospholipids
	27	Deodorization (chem. ref.) design 2	Causes degradation of phospholipids
	28	Deodorization (chem. ref.) design 3	Causes degradation of phospholipids
Protein separation	34	Protein concentration	
Microcomp. Refining	37	FADD separation	
	39	Tocos separation	
	41	Sterol separation	
Phosph. Refining	29	Lecithin dryer	
Protein/Fiber refining	32	Low-pro meal dryer	
	36	Low-pro Meal digestibility enhancement	
Byproducts processing**	43	Spent Clays valorization	
	44	Soapstock splitting	

Table 9.6 shows, for example, that interval 8 (hexane extraction) requires a chemical (hexane) as solvent. Hexane is an explosive chemical; therefore the need for ATEX design is identified. Moreover, it is underlined that this treatment leaves a residual concentration of water and hexane in the meal, which exceeds regulation limits. Therefore, the need for additional meal treatment to remove the residual water and hexane is identified at this stage. To resolve this issue, 2 additional process intervals (31 and 32, respectively low-protein meal dryer-cooler and high-protein meal dryer-cooler) are added in the lipid separation step. Also, this engineering insight result on the definition of a synthesis rule, stating that hexane extraction has to be followed by drying-cooling is generated (R-13 in table 9.7).
In the same way, intervals 18-22 represent an adsorption treatment (bleaching), requiring an adsorbent (clay or active carbon). The spent adsorbent cannot be regenerated, so those intervals are characterized by a consumption of fresh bleaching clays and a production of exhaust adsorbent, which need to be addressed. While the sourcing of clays does not represent an issue (they can be bought from a supplier), the disposal of exhaust clays is problematic, especially since the spent clays are flammable and cause explosion risk. Traditional solution for spent clays disposal used to be mixing with low-protein meal or land-filling (Thoenes, 2009), but both options are now excluded due to regulatory reason (European Commission, 2011).
As a consequence, spent clays need to be disposed as special waste (by paying a waste disposal price) or can be valorized through additional processing. In this work, co-firing of spent clays to produce steam in a boiler is considered as spent clay valorization option.
Similarly, soapstocks are byproducts of interval 14 (caustic degumming). Such a stream, containing a mixture of soaps and acid water, needs to be disposed or allocated to a product. A common employed technique is soapstock splitting (Thoenes, 2009), which consists of medium-high temperature separation of the soaps from the water. The former can be sold as a low value feed ingredient, while the latter is disposed as wastewater. In this study, soapstock splitting is added as a process interval.
As a consequence of the secondary flow analysis performed at this step, a byproduct processing step is added to the superstructure. In this step, all treatments necessary to handle byproducts and special waste streams are added. Those include spent clay valorization in a co-firing furnace (interval 43) and soapstock splitting (interval 44), and the respective products are added to the original product list defined in step 2.
Finally, engineering, commercial and regulatory insights (as well as considerations from the secondary flow analysis) are used to define new synthesis rules, at process interval level. Those synthesis rules complement the rules defined in step 2, and are reported in table 9.7.

Step 7: Process intervals superstructure generation The superstructure is generated, by adding the process intervals defined in table 9.6 as alternatives to execute each of the identified process steps.

Step 8: Process intervals superstructure screening The alternatives contained in the superstructure are screened against the synthesis rules listed in table 9.4 - 9.7. As a result, the superstructure is obtained, together with a list of logical

Table 9.7. Synthesis rules - process interval level

Rule	Description
R-8	Pressed oil (P1) can be produced only if lipid separation is performed via expelling (interval 5)
R-9	High-pro meal (P13) and Protein concentrate (P14) can be produced only if dehulling (3) is performed
R-10	FFADD, tocos and sterols (P-37-41) can be produced only if physical refining (11-13, 15-16, 19-20, 23-25) is used for their separation
R-11	Physical refining (11-13, 15-16, 19-20, 23-25) and chemical refining (14, 17-18, 21-22, 26-28) cannot be mixed
R-12	Expelling (5) must always be followed by hexane extraction (8,9)
R-13	Hexane extraction (8,9) should always be followed by meal dryer cooler (31,32)

constraints.
The superstructure is reported in figure 9.4, where bypasses are represented by a dashed line, and the correspondence between the process steps and the intervals superstructure is highlighted. Moreover, rule R-9, is translated into constraint 9.1, and rule R-5 into constraint 9.2.

$$y^3 + y^{32} \leq 1 \tag{9.1}$$

$$y^{45} + y^{52} \geq 0 \tag{9.2}$$

Where y^k is a binary variable which is 1 if interval k is selected. The superstructure for soybean processing generated through the superstructure synthesis method is composed of 65 process intervals: 2 raw materials (soybean of different quality), 42 process intervals (organized in 15 process steps), 20 potential products and 1 special waste (exhausted bleaching clay). Such a superstructure defines a search space containing more than 1 million potential networks.
The list of components includes 24 process components and 11 utility components. The superstructure generation task has been completed, and the problem formulation workflow continues with the data collection and model definition sub-step.

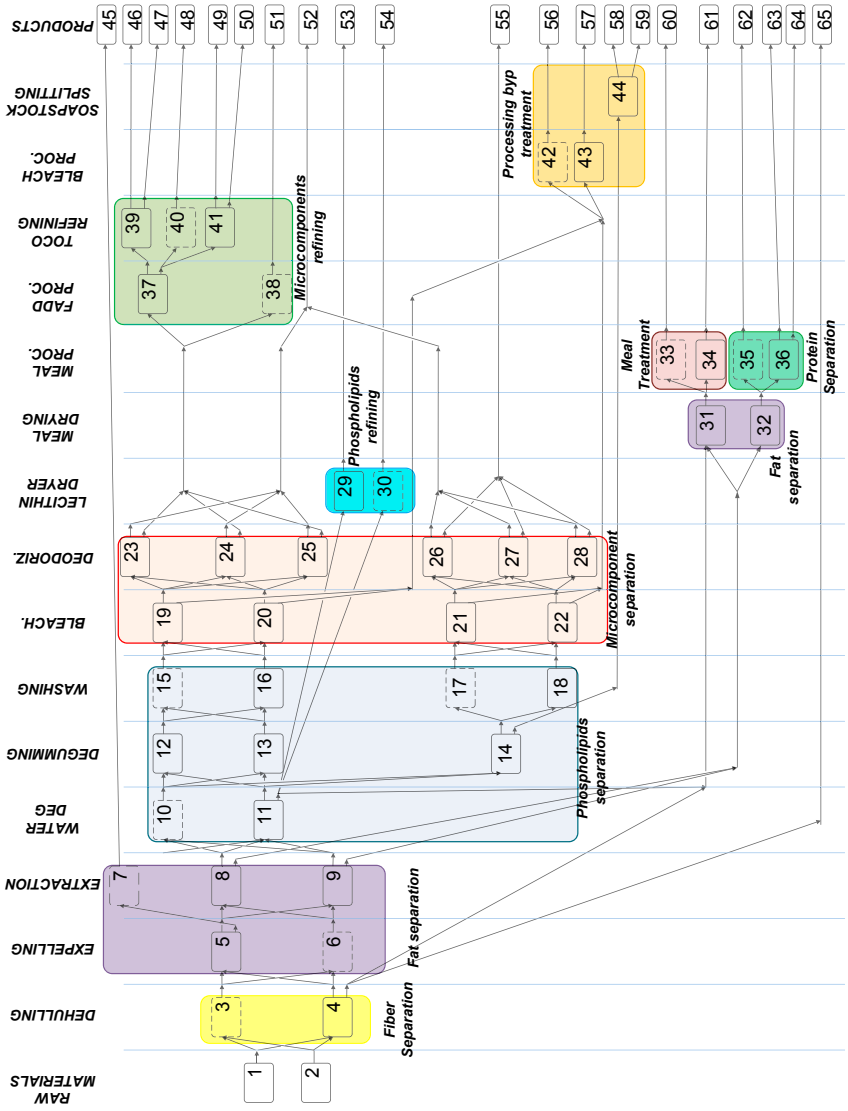


Figure 9.4. Superstructure for soybean processing. Dashed lines indicate bypasses

Data collection and model definition

The generic process interval model is used for the description of the process intervals contained in the superstructure, and the GOI model presented in chapter 3 (equation 3.35) is used as objective function. Since the calculation of GOI does not require the capital cost associated with the process intervals, equation 3.34 and 3.33 are eliminated from the model, resulting in the formulation of a MILP problem.
All model parameters such as raw material compositions, material prices, etc. are obtained from data available in the open literature or through Alfa Laval expertise. All secondary model parameters such as specific consumption of utilities or separation factors are calculated by solving the energy balance or the needed constitutive equations before solving the optimization problem.
As stated in the introduction, 4 scenarios are defined with respect to market prices and transportation distances:

- Scenario 1 (Base case): the average value of raw material, utilities and products prices is considered.
- Scenario 2: Low protein meal penalty; the market price of one potential product (low-pro meal) is reduced by 15%. This condition represents a common fluctuation of the market driven by the supply and demand of competing protein feed sources as gluten.
- Scenario 3: Transportation penalty; the market of some products (lecithin, tocopherols) is far from the production location, and the transportation cost is included.
- Scenario 4: Value of information; in this scenario the economic performances of the processing network calculated for scenario 1 is tested in the price conditions of scenario 2, in order to assess the economical importance of selecting the right processing network for a given market condition.

Since the soybean solution 1 is a deterministic problem, step 2,4,5,6 and 7 of the workflow are not executed.

9.1.2 Step 3: Deterministic solution

The deterministic MILP model is formulated and solved in GAMS using the CPLEX solver version 12 (IBM Corp., 2009). The model and solution statistics are reported in table 9.8 for the base case scenario. The results for the 4 different scenarios are summarized in table 9.9, where all economic indicators are reported in specific terms, referred to the processing of 1 unit-mass of soybean.
Together with each solution, a large amount of information such as stream table, utility table, waste emission, cost breakthrough, etc. are obtained as results. This information provide additional knowledge and allow deeper understanding of the financial and engineering dimension of the problem. Moreover, they constitute a good starting point for the further steps of project development, such as the preparation of Front-End Engineering Documentation (FEED), permitting, etc. In order to give an idea of the type of results obtained and of their relevance, additional results are

Table 9.8. Soybean processing solution 1 - solution statistics

Problem	Base case
n binary variables	64
n constraints	159,250
Relative optimality tolerance	1.00E-06
CPU time (s)	6.11
Solution algorithm	direct
n outer iterations	-
Average CPU time for 1 iteration (s)	-

reported in appendix D.
The results obtained for each scenario have been compared with the industrial standards, and critically discussed with the experts of Alfa Laval. The results of the analysis showed that the framework was able to identify solutions in line with what the experts expected for each scenario. On this basis, the results of the case study have been considered qualitatively validated.
In the next section, the results obtained for each scenario are briefly discussed.

Scenario 1(base case)

For scenario 1, which represents the base case, the optimization selects a product portfolio consisting of Fatty Acid Distillates (FFADD, process interval number 51), refined oil (52), lecithin (53) and low protein meal (60). Because of the high value of lecithin, the high quality soybean (2) is selected as raw material to have a high lecithin yield due to the higher content of hydratable phosphorous in the raw material.
Since low protein meal is selected in the protein portfolio, hulls separation from the solid stream is not necessary and therefore dehulling is bypassed (3). Mechanical separation of oil by expelling is also bypassed (6), as usually happens for low oil yielding seeds such as soybean.

Table 9.9. Soybean processing solution 1 - comparison of the results between the 4 scenarios

Indicator (specific)*	**Unit**	**Scenario 1**	**Scenario 2**	**Scenario 3**	**Scenario 4**
Cost	unit-cost	40.92	40.91	40.76	40.92
Raw material cost	unit-cost	40.25	40.25	40.14	40.25
Utility cost	unit-cost	0.67	0.66	0.63	0.67
Transportation cost	unit-cost	-	-	-	-
Special waste cost	unit-cost	1.52E-03	1.51E-03	1.52E-03	1.52E-03
Revenues	unit-cost	45.65	45.41	45.22	41.47
Oil revenues	unit-cost	15.94	15.87	15.91	15.94
By-products revenues	unit-cost	29.71	29.54	29.30	25.53
Input-to-product yield	%	93.09%	92.02%	92.11%	93.09%
Input-to-waste yield	%	6.91%	7.98%	7.89%	6.91%
GOI	unit-cost	4.73	4.50	4.45	0.55
GOI variation**	%	-	-4.80%	-5.70%	-88.40%
Topology		**Scenario 1**	**Scenario 2**	**Scenario 3**	**Scenario 4**
Raw material		2	2	1	2
Process path		3,6,8,11,12,15,20, 25,29,31,33,38,42	4,6,8,11,12,15,20, 25,29,32,35,38,42	4,6,8,10,12,15,20, 25,32,35,38,42	3,6,8,11,12,15,20, 25,29,31,33,38,42
Product portfolio		51,52,53,56,60	51,52,53,56,62	51,52,56,62,65	51,52,53,56,60

* all economic indicators are specific to 1 unit-mass of soybean processed.

** with respect to the value of the base case solution (scenario 1).

Oil extraction is performed in an energy integrated crush process (8) in order to reduce steam and cooling water consumption. In order to produce soy lecithin, hydratable phosphatides are removed by water degumming (11), and stabilized by drying (29).
The oil purification follows the physical refining route, as normally done for soybean processing in order to limit the wastewater load and to allow a better valorization of the fatty acid via the production of FFADD. The oil is acid degummed (12) to reduce the phosphorous concentration to specification level. No exhaust bleaching clay valorization process is selected (42), therefore the amount of bleaching clays is minimized by employing 2 stage bleaching (25). Deodorization is performed in a thin film deodorizer (28). Soybean meal is dried (31) in order to meet moisture specifications for low protein meal and sold without any further processing (33). The resulting specific GOI is 4.73 unit-cost per unit-mass of processed soybean. In table 9.10 the stream table describing the material flow through the network for this solution is reported.
For the above described considerations, the synthesized processing network can be considered qualitatively validated, being in agreement with the current industrial standards (Thoenes, 2009).

Scenario 2: low protein meal penalty

In scenario 2 a price penalty is applied to the price of low protein meal. As expected, the optimizer responds to this penalty by switching from low protein meal (60) to high protein meal (62) in the product portfolio. To achieve higher protein concentration in the meal, soybeans are pre-processed by dehulling (4) in order to remove and segregate the hulls, which are rich in fibres. As a consequence, hulls (65) are produced and therefore are added to the product portfolio. The rest of the processing network is equivalent to scenario 1, but some differences in the overall performance can be observed as a result of dehulling. The overall waste production is increased, because of the fine particles which are lost in the air classification step of the dehulling. Also, because of the oil entrapped in the hulls, the overall oil yield is reduced by nearly 0.5%.
Both the new synthesized processing network and the above discussed changes in the overall mass balance are judged reasonable and in line with industrial standards. The effect of the low protein meal penalty is the reduction of the GOI by 4.5% with respect to the base case.

Scenario 3: transportation penalty to access pharma markets

In this scenario the access to lecithin market is more difficult because of transportation costs. As expected, lecithin production becomes not economically convenient, and consequently is excluded from the product portfolio. Consequently, the premium paid for the high quality raw material (for higher hydratable phosphorous concentration) is not justified, so the optimizer switches to the selection of the less expensive raw material (1). Water degumming is bypassed (10) and all phosphatides

Table 9.10. Soy processing - deterministic solution, base case stream table

$F_{IN}^{i,k}$	3	6	8	11	12	15	20	25	29	31	33	38	42	51	52	53	56	60
TAG	20.78	20.78	20.78	20.21	20.15	20.15	20.15	20.09	0.06	0.58	0.58	0.20	0.06	0.20	19.89	0.06	0.06	0.58
DAG	0.11	0.11	0.11	0.10	0.10	0.10	0.10	0.10		0.00	0.00	0.00		0.00	0.10			0.00
MAG	0.11	0.11	0.11	0.10	0.10	0.10	0.10	0.10		0.00	0.00	0.00		0.00	0.10			0.00
FFA	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15		0.00	0.00	0.15		0.15				0.00
HP	0.07	0.07	0.07	0.06					0.06	0.00	0.00					0.06		0.00
NHP	0.66	0.66	0.66	0.64					0.32	0.34	0.34					0.32		0.34
Water	12.00	12.00	12.00							11.99	5.70							5.70
Tocopherols	0.04	0.04	0.04	0.04	0.04	0.04	0.04			0.00	0.00	0.04		0.04	0.00			0.00
Tocotrienols	0.01	0.01	0.01	0.01	0.01	0.01	0.01			0.00	0.00	0.01		0.01			3.0E04	
Sterols	0.07	0.07	0.07	0.07	0.07	0.07	0.07			0.00	0.00	0.07		0.07				0.00
Chlorophylls	2.2E4	2.2E4	2.2E4	2.1E4	2.1E4	2.1E4				6.1E6	6.1E6		2.1E4			2.1E4	6.1E6	6.1E6
Metals	2.2E4	2.2E4	2.2E4	2.1E4	2.1E4	2.1E4				6.1E6	6.1E6		2.1E4			2.1E4	6.1E6	6.1E6
Pheophytins	2.2E4	2.2E4	2.2E4	2.1E4	2.1E4	2.1E4				6.1E6	6.1E6		2.1E4			2.1E4	6.1E6	6.1E6
Color	2.2E4	2.2E4	2.2E4	2.1E4	2.1E4	2.1E4				6.1E6	6.1E6		2.1E04			2.1E4	6.1E6	6.1E6
Protein	18.00	18.00	18.00							18.00	18.00						18.00	18.00
Fiber	32.00	32.00	32.00							32.00	32.00						32.00	32.00
Sugars	16.00	16.00	16.00							16.00	16.00						16.00	16.00
Process-water					3E-4													0.48
Hexane										0.48	0.48							0.00
Bleaching-earth										0.05	0.04							
													0.21				0.21	

are removed as waste by acid degumming (12), causing an increase in waste production. The rest of the processing network is equal to the result of the previous scenario.
As a consequence of the transportation penalty, a reduction of GOI by 5.7% is observed.

Scenario 4: value of information

In this scenario the economical performances of the processing network synthesized in scenario 1 are tested against the market conditions of scenario 2, in order to assess the importance of processing network synthesis and design on profitability.
Since all the strategic decisions (raw material selection, selected processes and product portfolio) are fixed, no strategy can be implemented to react to the 15% reduction of low protein meal market price, and from a process engineering perspective the process performs equally to scenario 1.
From a business perspective, on the other hand, a big difference in performance is observed. The drop of low protein meal price in fact causes a reduction of the overall revenues by 9.3%. Soybean processing being a low margin operation, the reduction in revenues is amplified in terms of margin, where the GOI shows a dramatic 88.4% drop.

9.2 Soybean processing solution 2 (under uncertainty)

9.2.1 Step 1: Problem formulation

The formulation of *Soybean processing solution 2* is based on a modification of the formulation for the *solution 1*. With respect to the previous formulation, the objective function is modified to the maximization of Earnings Before Interests and Tax (EBIT), and the workflow for solution under uncertainty is selected. Moreover, in order to simplify the solution of the problem under uncertainty, the piecewise linearization of the capital cost constraints formulation is adopted, as described in note 1.

9.2.2 Step 2: Uncertainty domain definition

In the second step of the workflow, the domain of uncertainty is defined and characterized. The identification of the uncertain data is performed based on the information available in the open literature and on the expertise provided by Alfa Laval. As a result, 11 uncertain data are identified. These include 7 data related to market prices (2 raw material prices, 2 utility prices and 3 product prices), 2 composition data (protein concentration in the 2 feeds) and 2 process performance data (the steam consumption required for protein separation and the recovery of Non-Hydratable Phosphorous (NHP) in the water degumming process (11)).
Each of the uncertain data identified is characterized in term of probability dis-

tribution, based on available observations or on expert review. For some of the market-related data (standard quality soybean price, oil price, low-protein meal price) historical data related to the last 5 years have been retrieved from Alfa Laval. These data have been analyzed using standard statistical tools, in order to determine appropriate distribution function and correlation structure for these data, as reported in tables 9.11 and 9.12.
It is important to underline that using historical data for the characterization of future uncertainty implies assuming that the uncertainty domain related to the past will remain valid in the future.
For the remaining data, no experimental observations have been found. Consequently, the uncertainty characterization has been performed based on Alfa Laval expertise.
As a result, the domain of uncertainty described in tables 9.11 and 9.12 is defined. Monte Carlo sampling is performed, and 80 future scenarios with equal probability of realization are generated with respect to the value of the 11 uncertain data. The generated samples are represented in figure 9.5. From visual inspection, it can be seen that the samples generated reproduce the correlation of uncertain data defined in table 9.12.

Table 9.11. Soybean processing solution 2 - Uncertainty characterization

Data	Mean	Prob. distr.	Min	Max	Description
pi_R^1	350	Uniform	245	455	Low quality soy price
π_R^2	355	Uniform	248.5	461.5	High quality soy price
$\pi_U^{Steam10b}$	25	Uniform	18.75	31.25	Steam-10b price
π_P^{52}	900	Uniform	630	1170	Refined oil price
π_P^{60}	310	Uniform	155	465	Low-Pro meal price
π_P^{62}	320	Uniform	272	368	High-Pro meal price
π_P^{63}	900	Uniform	765	1035	Protein concentrate price
$\phi^{Prot.,1}$	38	Uniform	34.2	41.8	Protein content low quality soy
$\phi^{Prot.,2}$	38	Uniform	36.1	39.9	Protein content high quality soy
$\mu^{steam10,i,36}$	0.33	Uniform	0.27	0.4	Steam cons. protein sep.
$\sigma^{11,NHP}$	0.5	Uniform	0.43	0.58	Separation NHP water degumming

9.2.3 Step 3: Deterministic solution

The uncertain data are fixed at their expected value, and a MILP problem containing 154,000 constraints and 485 binary variables is formulated and solved in GAMS, using the CPLEX solver (IBM Corp., 2009).
As a result, the optimal network reported in figure 9.6 is identified for deterministic conditions. For this network, an EBIT value of 108.5 unitcost per unitmass of process soybean per year is calculated.
It is interesting to compare this result with the base case solution obtained in soybean processing solution 1 (reported in table 9.9). From this comparison, it can be observed that 2 different configurations are obtained for the same conditions, for different objective functions (GOI for *solution 1*, EBIT for *solution 2*).

Table 9.12. Soybean processing solution 2 - correlation matrix of the uncertain data

	SoyBean Low Quality price	SoyBean High Quality price	Steam-10b price	Refined oil price	LowPro Meal price	High Pro Meal price	Protein concentrate price	Protein content low quality	Protein content high quality	Steam Protein Separation	Separation NHP water degumming
SoyBean Low Quality price	**1**	0.58	0	0.56	0.3	0.38	0.53	0.38	0	0	0
SoyBean High Quality price	0.58	**1**	0	0.41	0.35	0.32	0.58	0	0.56	0	0
Steam-10b price	0	0	**1**	0	0	0	0	0	0	0	0
Refined oil price	0.56	0.41	0	**1**	0.78	0.66	0	0	0	0	0
LowPro Meal price	0.3	0.35	0	0.78	**1**	0.74	0	0	0	0	0
High Pro Meal price	0.38	0.32	0	0.66	0.74	**1**	0	0	0	0	0
Protein concentrate price	0.53	0.58	0	0	0	0	**1**	0	0	0	0
Protein content low quality	0.38	0	0	0	0	0	0	**1**	0	0	0
Protein content high quality	0	0.56	0	0	0	0	0	0	**1**	0	0
Steam Cons. Protein Separation	0	0	0	0	0	0	0	0	0	**1**	0
Separation NHP water degumming	0	0	0	0	0	0	0	0	0	0	**1**

Figure 9.5. Soybean processing solution 2 - Sampling results. The uncertain data are reported as θ_1 - θ_{11} according to the order reported in table 9.11

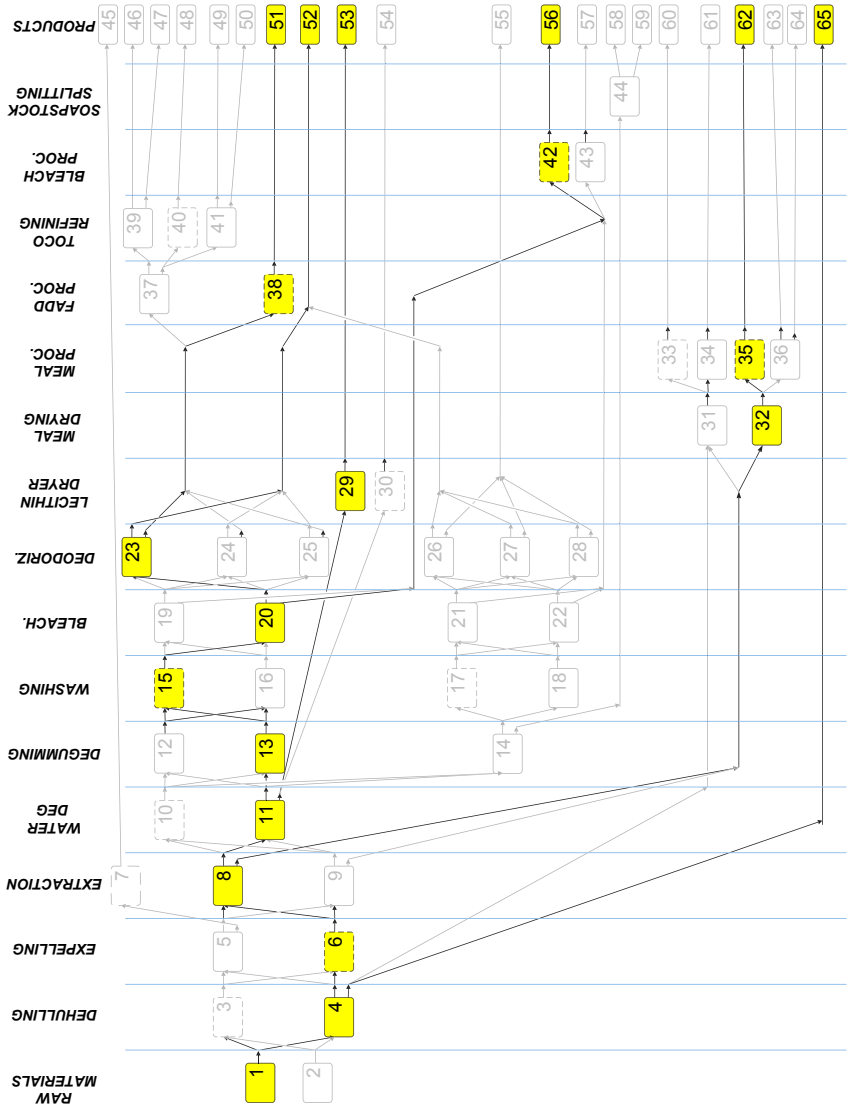


Figure 9.6. Soybean oil solution 2, Step 3 Results: Optimal processing network under deterministic conditions

9.2.4 Step 4: Uncertainty mapping

In this step, the consequences of the uncertainty on the design problem are mapped, by performing a separate deterministic optimization for all the future scenarios generated by Monte Carlo sampling in step 2.

The results (reported in figure 9.7) show that, for different realization of the un-

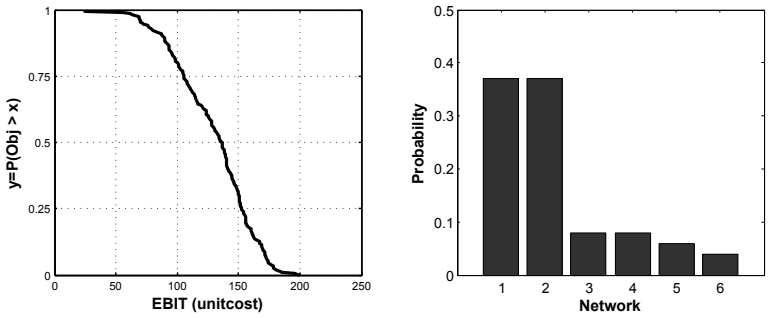


Figure 9.7. Soybean processing results - uncertainty mapping results

certain data, 6 different processing networks are identified as optimal. The domain of uncertainty defined in step 2 exhibits a significant impact on the performances of the design, which is confirmed by the large variability which is observed for the objective function, with EBIT values spanning from 25.2 to 200.1 unitcost/yr, as reported in the left hand-side of the figure.

The analysis of the network configurations obtained for the different scenarios indicate that that 13 intervals (of 65) are part of all optimal networks which are identified, while 39 are never part of one.

These considerations indicate once again that the defined domain of uncertainty has a significant impact on the design problem and on the performances of the processing network.

9.2.5 Step 5: Solution under uncertainty

The problem of decision-making under uncertainty is formulated as described in chapter 4, resulting in a MILP equivalent with more than 10 million constraints.

In order to reduce the size of the problem, the superstructure reduction policy and the variable bounding strategy described in chapter 5 are implemented, resulting in the formulation of a tractable simplified problem, with more than 1 million constraints. Since the conditions reported in note 1 are not met, the simplification performed is not an exact method, and therefore there is no proof that the solution of the simplified problem corresponds to the global optimum of the original problem. The simplified problem is solved via the bi-level decomposition presented in chapter 5.2.2.

As a result, the network reported in figure 9.8 is identified, and a corresponding expected EBIT of 106.8 unitcost/yr is calculated.

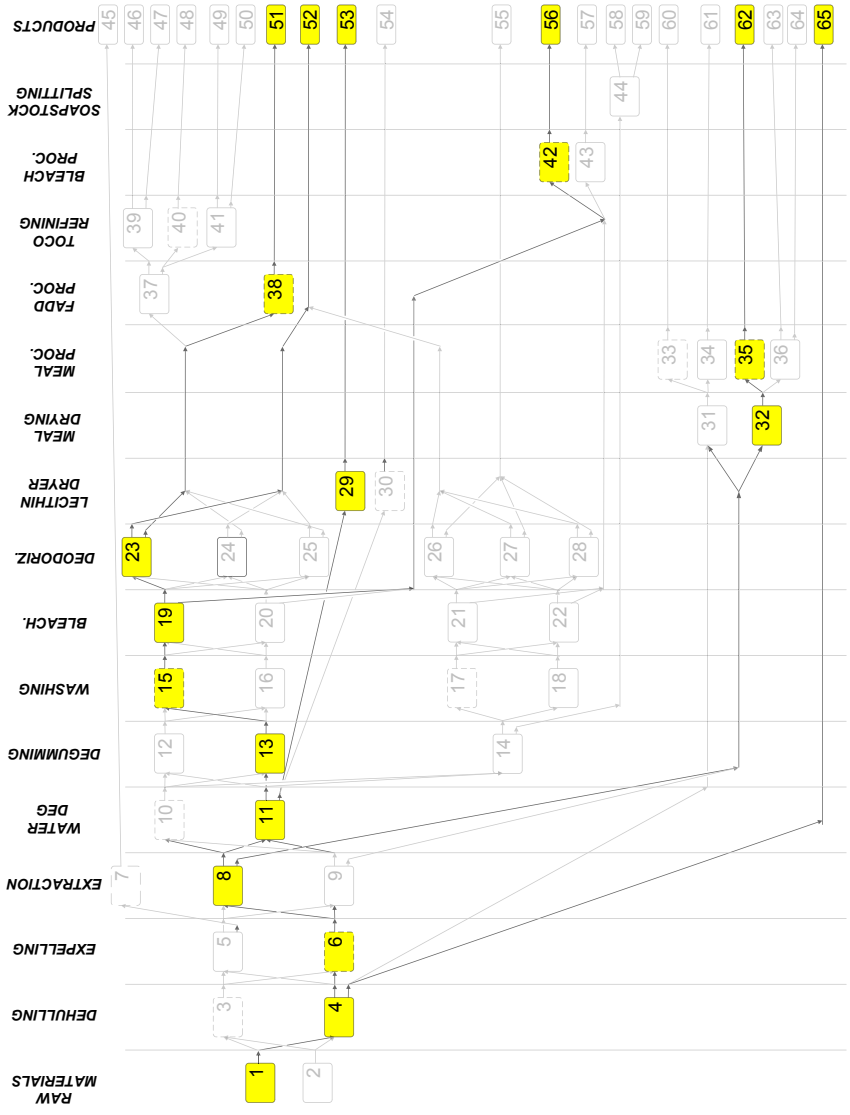


Figure 9.8. Soybean processing solution 2 - Step 5 results: Optimal processing network in uncertain conditions

9.2.6 Step 6: Optimal flexible solution

The flexible network problem is formulated as described in chapter 4, for the reduced superstructure obtained through the superstructure reduction policy, resulting in a MILP problem with more than 12 million constraints.
The problem is solved using the bi-level decomposition strategy described in section 5.2.2. Because of the size and complexity of the problem, the convergence criterion is set to accept a gap of 10%.
The solution algorithm stops because the convergence criterion is satisfied after 16 iterations and a total computational time of 29 hours. Problem statistics are reported in table 9.13.
Although large, the computational time required for the solution is deemed accept-

Table 9.13. Soybean processing solution 2 - solution statistics

Problem	**Optimal flexible network**
n binary variables	485
n constraints	12,296,000
Relative optimality tolerance	8.3%
CPU time (s)	106,360
Solution algorithm	bi-level decomp.
n outer iterations	16
Average CPU time for 1 iteration (s)	4,432

able for a design problem. Further reduction of the computational time can be obtained by using a more powerful computer.
Through its redundant structure, the obtained network configuration (reported in figure 9.9) has the ability of shifting between the production of low-protein meal (interval 60) and the production of high-protein meal and hulls (intervals 62 and 65), depending on the realization of the uncertain data.
Our industrial partner Alfa Laval has confirmed that this strategy is indeed in agreement with the consolidated industrial practice, to respond to market prices and raw material quality fluctuations. This observation constitute a validation, at least qualitative, of our method and solution strategy.
It is important to underline that since the superstructure reduction policy has been used to simplify the problem, this solution is not guaranteed to be the global optimum of the original design problem.
The financial performances of this solution, nevertheless, appears extremely positive if compared with the performances of the optimal network under uncertainty, with an expected EBIT calculated as 115.3 unit-cost/year, corresponding to an improvement of 6.2%, with respect to the solution under uncertainty.
The high performance of the flexible solution highlights how, even though the superstructure reduction policy may lead to the identification of a local solution, this solution may still have a great value in terms of industrial decision-making, as observed for example in this case study.

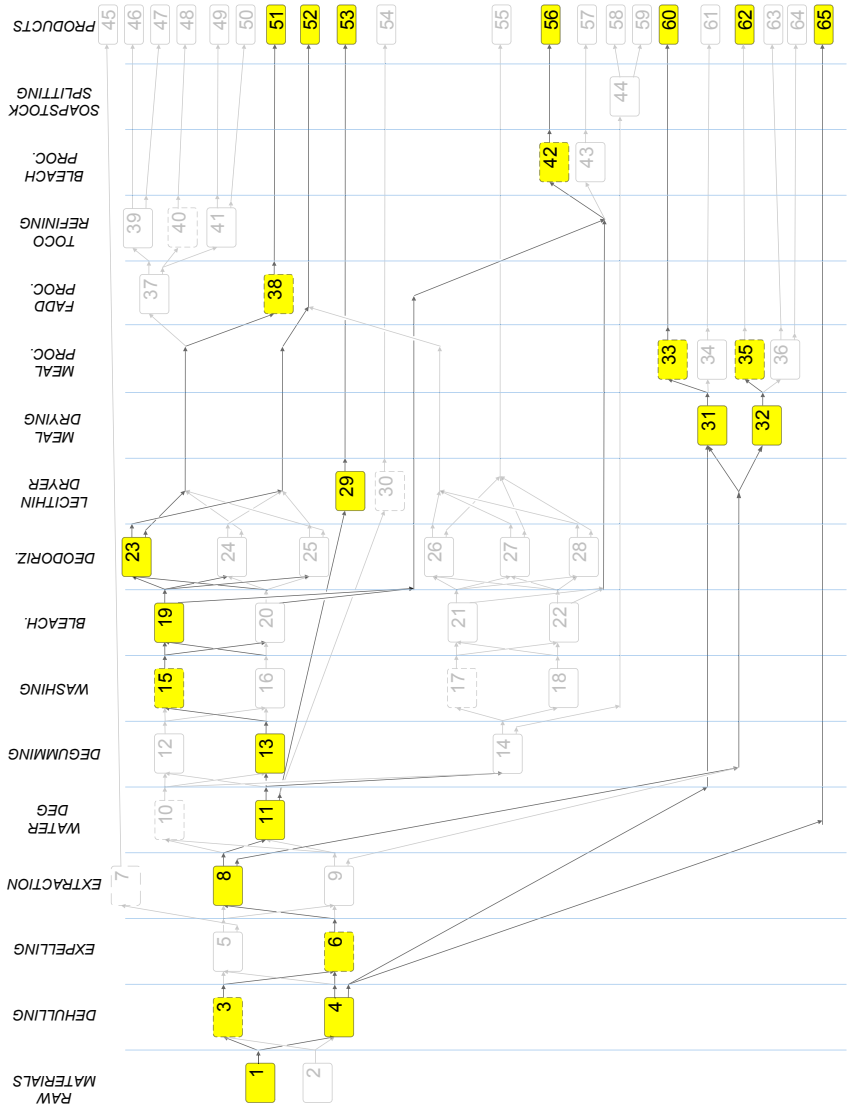


Figure 9.9. Step Results: Optimal flexible processing network in uncertain conditions

9.2.7 Step 7: Report generation

In the last step of the workflow, a report aggregating all the obtained results is generated. The summary of the report is given in table 9.14.
From the analysis of the results, it can be observed that 3 different networks are identified as optimal configurations in deterministic conditions, under uncertainty and as optimal flexible solution.
Furthermore, a large EVPI value is calculated (19.3 unitcost/yr , 17.7% of EBIT), indicating that a considerable performance increase could be obtained, if a "more certain" knowledge of problem data could be developed. Since the majority of the uncertain data are market prices, this result underlines the importance of having good market forecast in order to design of profitable soybean processing networks.
The other indicators confirm the outstanding financial performances of the identified flexible network, underlining how the flexibility in the product portfolio allows adapting to the variations of the business environment, in order to be able to capture the highest value from the products for each scenario. This is reflected by the large value of VSS (8.8 unitcost/yr, 8.1% of the EBIT), and by the negative value of UP (-6.7 unitcost/yr). The interpretation of the negative value of UP is that, because of its flexibility, the proposed network can actually benefit from price fluctuations. From these results, the flexible network appears as the optimal decision.

9.3 Conclusions from the soybean processing case study

In this case study, the application of the framework to a problem of industrial complexity has been demonstrated.
In particular, through the *Soybean processing solution 1*, the method for systematic synthesis of processing network superstructures presented in chapter 6 has been demonstrated on a large scale problem, and its capability of allowing transparent definition of the search space for optimization-based design methods has been highlighted.
Through the solution of the design problem for different scenarios, the flexibility of the tool has been proven. Moreover, by analyzing the different network structures obtained under different scenario definitions, a qualitative validation of the framework and of the results has been obtained. This contributed to build the trust of Alfa Laval's experts in the usefulness of the developed framework.
With the explicit consideration of uncertainty performed in *solution 2*, the ability of the framework to handle large and complex design problems under uncertainty has been demonstrated.
First, the consequences of market, supply-chain and technical uncertainty on the design of a large supply chain have been tested. Then, the uncertainty has been included in the decision-making problem, leading to the identification of an extremely convenient flexible solution. Such a solution, in fact, constitutes the optimal trade-off between capital investment and operational flexibility, for a defined domain of uncertainty. Through the ability of switching operational policy and product portfo-

lio depending on the market conditions, such a configuration allows capturing more value in uncertain conditions than in the deterministic case.
Furthermore, this solution indicated the importance of solution benchmarking, which is allowed by the framework structure. Because of the step-wise structure of the workflow of problem solution, in fact, the design problem is solved multiple times, each time increasing the complexity related to the consideration of data uncertainty in the problem formulation. As a consequence, simplified versions of the problem are solved prior to consider the overall complexity of the stochastic problem.
These intermediate results can be used in order to benchmark the solution of the stochastic problem. This feature is particularly important especially when the stochastic problem proves to be too complicated to be solved to global optimality, since it allows evaluating the quality of the local solution obtained.
In the case presented here, for example, the local solution obtained for the stochastic problem shows better performances than the global solution obtained under deterministic conditions, and therefore it can be concluded that such a solution, even if local, has a considerable industrial value. The validity of this result has been confirmed by edible oil industry experts, who recognized in it an established industrial practice to manage market uncertainty.
From a numerical point of view, the ability of the framework and of the solution methods employed to handle problems of considerable size and complexity has been explored. The pragmatic simplification methods presented in chapter 5 have proven the ability to reduce the problem to a tractable form, even though the solution of the simplified problem still required 29 hours of computational time. The global optimality of such a flexible solution could not be proven, because of the use of the superstructure reduction policy, and of the high relative optimality tolerance obtained. Nevertheless, by benchmarking its performance with the one obtained for deterministic conditions, it can be concluded that the obtained solution has a great value from an industrial perspective.

Table 9.14. Soybean processing - results report

Solution	**Network**	**EBIT (unitcost/yr)**
Optimal network	1 4 6 8 11 13 15 20 23 29 32 35 38 42 51 52 53 56 62 65	108.5
Network u/ uncertainty	1 4 6 8 11 13 15 19 23 29 32 35 38 42 51 52 53 56 62 65	106.8
Flexible network	1 3 4 6 8 11 13 15 19 23 29 31 32 33 35 38 42 51 52 53 56 60 62 65	115.3

Indicator	**VSS (unitcost/yr)**	**UP (unitcost /yr)**	**EVPI (k$/yr)**
Network u/ uncertainty	1.8	0.3	19.3
Flexible network	8.8	-6.7	

10

Oil Refinery Wastewater Treatment and Reuse

The third and last case study reported in this thesis is related to the synthesis and design of a network for the treatment and reuse of oil refinery wastewater. This case study aims at demonstrating the features of the framework with respect to the problem of optimization-based design of water networks, which constitutes a relevant application area.

Optimization-based design of industrial wastewater treatment and reuse networks

Among the wide variety of process synthesis and design problems, the problem of synthesis and design of industrial wastewater networks represents a particularly interesting example, because of its scope and significance, as well as because of the technical challenges which it contains.
Water is in fact a valuable resource of great relevance for humanity, and the awareness of the importance of its conservation has dramatically increased over the last decades. The total volume of water present on Earth amounts to $1.386 \cdot 10^9 km^3$, of which only 2.5% is freshwater, out of which 30% embodied in glaciers. In 2010, the global freshwater withdrawal amounted to around $3.9 \cdot 10^3 km^3/yr$, 60% of which for agricultural use, 22% for domestic use, and 18% for industrial use (Gleick, 2012). Prior to discharge to the environment, large part of the water withdrawn for domestic and industrial use needs to be treated, in order to reduce its contamination level. The design of water treatment and reuse systems is therefore crucial to ensure cost-effective sustainable use of water.
The goal of a wastewater treatment and reuse network is the reduction of the pollutant load in a wastewater stream, to a level which allows discharge in the environment or reuse in a water-using process. Wastewater treatment processes are generally composed by three stages: a primary treatment, based on physical operations to remove non soluble suspended solids; a secondary treatment for removal of dissolved contaminants through chemical or biological processes; and a tertiary treatment for removal of residual contaminants. A complete and descriptive overview of technologies and issues related to wastewater treatment operations can be found in Eckenfelder *et al.* (2009) and Tchobanoglous *et al.* (2003).
The configuration of a wastewater treatment plant is largely determined by the type and concentration of contaminants present in the wastewater, the effluent discharge limits to be met and the environmental conditions (temperature, precipitation).
In the industrial practice, the design of a wastewater treatment plants is an expert-

based procedure, which requires specific know-how and often involves laboratory and pilot-scale trials, and may result in a costly and time consuming task (Wong and Hung, 2004). Therefore, because of the time and resources constraints that are typical of engineering projects, it is often possible to evaluate only a small number of alternatives with respect to treatment configuration and water recycle opportunities. This may result (especially for complex cases) in a sub-optimal design, in which options for water recycle and/or recovery and valorization of contaminants may be disregarded.
Because of the relevance of the problem and of the scientific challenge which it represents, the problem of water network design has attracted the attention of many authors in the scientific literature, starting from the seminal work of Takama *et al.* (1980), who solved a water network for a refinery including both water using processes and water treatment processes.
More recently, Tan *et al.* (2009) developed a superstructure based approach for the synthesis of a water network, based on partitioning regenerators, considering a single contaminant and 4 wastewater streams.
Khor *et al.* (2011, 2012) studied the optimal design of a membrane based treatment for the treatment and reuse of water contaminated by a single contaminant, for a system of 28 wastewater sources, 15 water sinks and 17 wastewater treatment.
Karuppiah and Grossmann (2006, 2008) proposed a spatial branch-and-bound algorithm for the solution of the water network problem, and applied it to a case including both the water using and the water treatment section, considering 3 contaminants, 5 water user and 4 treatment options.
Bogataj (2008) formulated and solved the problem of synthesis of heat integrated water networks, in which the synthesis and the heat integration problem are solved simultaneously, for a system of 3 contaminants and 5 treatment options.
Rojas-Torres *et al.* (2013) investigated the design of a water network based on property integration, for a system of 6 wastewater flows and 15 treatment options, considering one contaminant and 4 temperature dependant properties (pH, Toxicity Index, odor and temperature).
Teles *et al.* (2012) proposed an algorithm based on parametric disaggregation for the solution of non convex MINLP problems featuring bilinear terms, and demonstrated it by solving a number of optimization problems, including water networks.

Treatment and reuse of oil refinery wastewater networks

The problem of water management in oil refineries represents a challenging case study with considerable industrial relevance. Oil refineries, in fact, are characterized by an intensive use of water (consuming between 1.55 and 2.14 m^3 of water per m^3 of crude oil), and are often located in water-scarce geographies (Wong and Hung, 2004).
Typical water uses in oil refinery include cooling, steam generation and washing. Additionally, water is also used in the desalter to remove solids and salts from the crude prior to sending it to the crude distillation units (Arena and Buchan, 2006).
Refinery effluents are highly variable in flow and composition. In general, the range of contaminants contained in wastewater streams includes hydrocarbons, dissolved

materials, suspended solids, phenols, ammonia, sulfides and metals (Eckenfelder *et al.*, 2009; Wong and Hung, 2004).
Because of the variability existing in flow and composition of wastewater originated by different refineries, and since different emission limits on water effluent apply in different geographies, the replication of a standard water treatment and reuse design for different plants may be unfeasible, or not represent the best alternative with respect to economical or environmental considerations.
Consequently, the developed framework appears to have the potential to contribute to the design of industrial wastewater treatment and reuse networks, by providing the ability of quickly screening among a large number of alternative network configurations (including water recycle options), in order to identify a reduced set of promising candidates, on which the more time consuming and expensive phases of design, such as detailed modeling and experimental verification can be focused.
For these reasons, the problem of oil refinery wastewater treatment and reuse design has been selected as a case study.
In order to demonstrate the capability of the framework in handling the challenges associated with this case study, the Wastewater Treatment and Reuse (WTR) problem will be solved for 2 cases, denominated *WTR solution 1* and *WTR solution 2*.
In *WTR solution 1*, the single-stream problem will be formulated and solved for uncertain influent composition, aiming at the minimization of the expected total annualized cost of the treatment plant.
In *WTR solution 2*, the multi-stream problem will be formulated and solved in deterministic conditions, in order to demonstrate the capability of the framework to formulate and solve large scale non-convex problems, leading to the identification of design including water recycle.

10.1 Oil refinery wastewater treatment and reuse solution 1 (single-stream)

10.1.1 Step 1: Problem formulation

Problem definition

A schematic representation of the problem is reported in figure 10.1. Given the water-using plant represented in the left hand-side of the figure, consuming a certain amount of fresh water (at a required purity level) and producing a certain amount of wastewater (at a given contamination level), the goal of the problem is to design a WTR network (represented in the right hand-side) which maximize the optimality indicator.
As stated in the introduction *WTR solution 1* is formulated as a single-stream design problem under influent composition uncertainty, aiming at the maximization of the expected Total Annualized Cost (TAC), over an investment horizon of 10 years.
The determination of the optimal flexible network has been considered out of scope, because of the technical challenges related to operating a wastewater treatment plant with a variable configuration. Therefore, step 6 of the workflow has not been

performed.

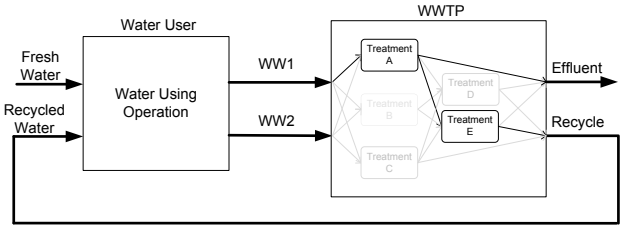


Figure 10.1. The optimal wastewater treatment plant design problem

Data collection and superstructure definition

The data needed for the formulation of the problem are obtained from the open literature.
In particular, based on the data reported by Eckenfelder *et al.* (2009) and Khor *et al.* (2011), the wastewater flows produced by an oil refinery have been characterized through the definition of 3 wastewater influent streams:

WW1: caustic wastewater representative of spent caustic from isomerization, alkylation and drying and sweetening.

WW2: sour wastewater representative of all sour wastewater sources (from distillation, cracking etc.).

WW3: oily wastewater representative of oily wastewater sources (not sour and not caustic).

The characterization of these streams in terms of flow and composition is reported in table 10.1.
Discharge as surface water is consider as an option for water sink. Moreover, based on the results reported by Arena and Buchan (2006), 3 water recycle options have been identified, leading to the identification of 4 potential water sinks alternatives:

D: discharge as surface water

DES: recycle as process water in oil desalter as process water

CW: recycle to cooling towers as water make-up

BFW: recycle as boiler feed water for the generation of steam.

For each option, maximum flow and contaminants concentration limits are specified based on environmental regulation and technical specifications, as reported in table 10.2.
The table shows how the identified recycle opportunities are characterized by different specifications in terms of water quality, depending on the water-user process. In particular, the desalter can admit a relatively high contamination level, while high

purity water is required by the steam boiler.
Consequently, it is reasonable to expect that the purification of the influent to a quality which allows recycling to the desalter will require less processing steps (and therefore less capital and operational investments) than what is required by the boiler feed water.
A water saving bonus is defined for each water recycle option (in terms of $/ton of recycled water), in order to account for the reduction in utility cost for the oil refinery process which derives from using recycled water instead of fresh water. The open literature is searched to identify known treatment configurations for the purification of refinery wastewater. The configurations reported in the literature are decomposed in sequences of treatments, each represented as process steps according to the ontology adopted in the framework. For each step, different technological alternatives are identified, and represented as process intervals. The list of process interval identified is reported in table 10.3.
For the sake of simplicity, the sludge treatment line has not been explicitly considered in the scope of the design problem; the costs related to sludge treatment (e.g. through anaerobic or aerobic digestion and ultimate disposal to landfill) are considered through the definition of a sludge disposal price.
As a result, the superstructure for oil refinery wastewater treatment and reuse showed in figure 10.3 is defined.
Each treatment unit is modeled using the generic process interval model described in chapter 3. An extensive description of the data collection and model parameters calculation step is given as appendix.
As described in chapter 6.3, the characterization of wastewater contaminants is performed based on a modified ASM component list.
With respect to the ASM list, a number of extra contaminants specific to oil refinery wastewater (such as Cr^{6+}, Cr^{3+}, Fe^{2+}, Fe^{3+}, H_2S, CO, CH_4, etc.) have been added. As a result, a component list containing 44 components has been defined.
All results are reported according to the traditional pseudo-components character-

Table 10.1. Wastewater influent definition (Eckenfelder *et al.*, 2009; Khor *et al.*, 2011)

		Caustic WW1	**Sour WW2**	**Oily WW3**
Flow*	t/h	0.2	100	558.8
COD	mg/L	80,491.40	869	1,333.40
NH_4^+	mg/L	551.4	1462.9	79.4
H_2S	mg/L	14,512.90	1,553.00	55.2
Cr^{6+}	mg/L	5.8	0	28
$O\&G$	mg/L	5	281.4	1,475.80
FSS	mg/L	0	0	470.0
BOD	mg/L	2,176.50	660	712.7
TSS	mg/L	0	0	940.1

* COD: Chemical Oxygen Demand, BOD: Biological Oxygen Demand, FSS: Fixed Suspended Solids, TSS: Total Suspended Solids, O&G: Oil and Grease

Table 10.2. Treatment objectives characterization - limitations on the maximum flow rate and pollutant concentrations.

Description		**Surface discharge**	**Recycle as BFW**	**Recycle as CW**	**Recycle to desalter**	**Reference**
Flow	ton/h	-	198.5	104.6	50.00	(Khor *et al.*, 2011)
COD	mg/L	105.3	3.5	52.6	-	(World Bank, 1999), (Tchobanoglous *et al.*, 2003)
NH_4^+	mg/L	4.9	0.1	-	50	(Alex *et al.*, 2008), (Tchobanoglous *et al.*, 2003)
H_2S	mg/L	1.1	-	-	10	(World Bank, 1999), (IPIECA, 2010)
Cr^{6+}	mg/L	0.1	-	-	-	(World Bank, 1999)
Cr^{3+}	mg/L	0.4	-	-	-	(World Bank, 1999)
Ca^{2+}	mg/L	-	0.4	50.0	-	(Tchobanoglous *et al.*, 2003)
SO_4^{2-}	mg/L	-	-	200.0	-	(Tchobanoglous *et al.*, 2003)
Fe^{2+}	mg/L	-	0.3	0.5	-	(Tchobanoglous *et al.*, 2003)
Fe^{3+}	mg/L	-	0.3	0.5	-	(Tchobanoglous *et al.*, 2003)
Cl^-	mg/L	-	-	500.0	-	(Tchobanoglous *et al.*, 2003)
CO_3^{2-}	mg/L	-	120.0	24.0	-	(Tchobanoglous *et al.*, 2003)
$O\&G$	mg/L	10.0	25.0	25.0	-	(World Bank, 1999), Khor *et al.* (2011)
FSS	mg/L	15.0	5.0	100.0	-	(World Bank, 1999), Tchobanoglous *et al.* (2003)
BOD	mg/L	30.0	3.5	52.6	-	(World Bank, 1999)
TSS	mg/L	30.0	5.0	100.0	-	(World Bank, 1999), Tchobanoglous *et al.* (2003)

Table 10.3. Summary of the wastewater treatment units included in the network.

Step	**Process interval**	**ID**
Oxidation	Wet Air Oxidation	WAO
	Chlorine Oxidation	ChOx
Stripping	Sour Water Stripper	SWS
	H_2S Stripper	SS
	NH_3 Stripper	NS
	Air Stripper	AirS
Gravity separation	American Petroleum Institute Separator	API
	CPI/PPI Separator	CPI/PPI
Floatation	Dissolved Air Flotation	DAF
	Induced Air Flotation	IAF
Biological treatment	Trickling Filter	TF
	Rotating Biological Contactor	RBC
	Activated Sludge	AS
	Activated Carbon assisted AS	PACT
	Membrane Biological Reactor	MBR
Adsorption	Adsorpion on Granular Activated Carbon	GAC
Precipitation	Phosphorous Precipitation	PhPrec
	Metal Precipitation	MePrec
	Chromium Precipitation	CrPrec
Electrostatic separation	Ion Exchange	IE
	Electrodialysis	ED
Tertiary Filtration	Microfiltration/Ultrafiltration	MF/UF
	Nanofiltration/Reverse Osmosis	NF/RO

ization discussed in section 6.3, in order to facilitate their understanding. The conversion between the characterization methods is performed through equations 6.12 - 6.16.

10.1.2 Step 2: Uncertainty domain definition

As mentioned in the introduction, the composition of oil refinery wastewater is function of many different internal and external factors such as crude oil quality, process configuration and temperature. Therefore, it constitute a significant source of uncertainty for the design of the network.
In order to characterize the uncertainty associated with these data, the scientific literature has been searched to identify the minimum and maximum values reported with respect to contaminants load (reported in table 10.4). The table shows that a variation of several orders of magnitude exists between the data reported for some contaminants, confirming the variability mentioned in the introduction.
Based on the data reported in table 10.4, 13 influent composition data are identified as uncertain, and characterized as uniform distributions between the upper and lower bound defined in the table. Since no information could be found or calculated with respect to the correlation between these data, for the sake of the case study the distributions have been assumed to be uncorrelated.
The defined domain of uncertainty has been sampled, obtaining a list of 150 possible future scenarios with equal probability of realization with respect to wastewater composition.

Table 10.4. WTR solution 1 (single-stream) - Maximum and minimum values reported in the scientific literature for influent composition

WW Source	**WW1**		**WW2**		**WW3**	
Contaminant (mg/L)	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**
H_2S	0.2	48,500	19	4,320	1.5	121.6
NH_4^+	2.8	1,100	36.1	3,342.5	2.9	205.9
COD	302	364,100	935	1,530	450	4,774
$O\&G$	0	0	12.7	550	22.6	9,357.5
Cr^{6+}	0	0	0	0	0.3	121.6
TSS	0	0	0	0	200.5	4,781.5

10.1.3 Step 3: Deterministic solution

The deterministic formulation of the single-stream problem resulted in a MILP problem containing 51,567 constraints and 41 binary variables, which have been solved using CPLEX in 2.6 seconds.
As a result, a water treatment network with an annualized cost of 17.454 M$/yr was obtained.
The selected configuration (reported in figure 10.3) is composed by SWS as pretreatment, CPI/PPI and IAF as primary treatment, AS technology as secondary

treatment and MF/UF and NF/RO as tertiary treatment. After this treatment sequence the water meets the purity requirements, and is discharged to surface water bodies. The stream table for this treatment network is reported in table 10.5.
Since the problem is defined as single-stream, no flow splitting can occur along the treatment train. Consequently, none of the recycle options is selected, since none of them could receive the entire flow, because of the flow limitations reported in table 10.2.
For this case, therefore, the evaluation of recycle opportunities requires the solution of the multi-stream problem.

10.1.4 Step 4: Uncertainty mapping

For each of the future scenarios defined in step 2, a separate design problem is solved. The results are shown in figure 10.2. On the left hand-side of the figure, the cumulative distribution of objective function values shows that the TAC of the treatment process is extremely dependent on the influent wastewater composition.
The right hand-side of the figure shows that a large number of different network result optimal for different scenarios, even though one of them (network 1) is optimal for more than 60% of the samples.
More details with respect to the networks identified in the uncertainty mapping step are reported in table 10.6. From this table, it can be observed how some of the treatment steps are always bypassed. Moreover, it can be seen that the configuration obtained as solution of the deterministic case appears to be optimal for only 9% of the samples.

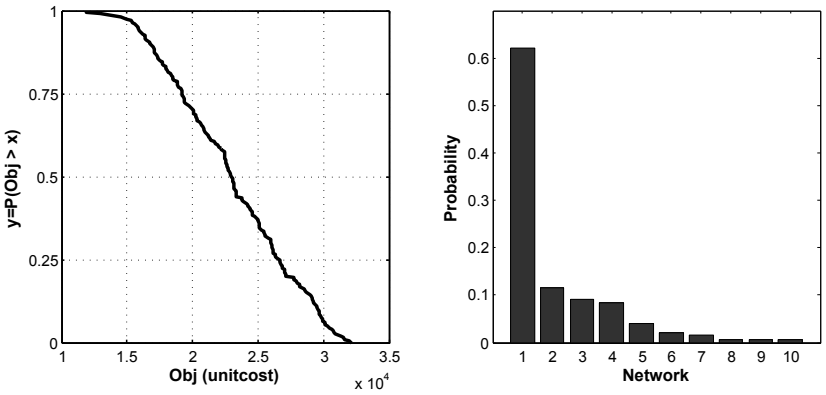


Figure 10.2. Oil refinery wastewater treatment and reuse solution 1 (single-stream) - Uncertainty mapping results

Table 10.5. WWTP Solution 1 (single-stream): Stream table for the deterministic solution

		Influent	**SWS**	**CPI/PPI**	**IAF**	**AS**	**MF/UF**	**NF/RO**	**D**	**limits**
COD	mg/L	1,289.88	1,289.88	748.13	486.29	486.29	35.2	24.21	24.21	105.3
Cr^{6+}	mg/L	23.76	23.76	23.76	23.76	23.76	2.5	0.08	0.08	0.1
H_2S	mg/L	287.49	0.11	0.11	0.11	0.11	0.13	0.13	0.13	1.1
NH_4^+	mg/L	274.92	49.49	49.49	49.49	49.49	39.54	1.24	1.24	4.9
$O\&G$	mg/L	1,297.05	1,297.05	129.7	9.08	9.08				10
FSS	mg/L	399.49	399.49	159.79	27.96	27.96	0.31	0.01	0.01	15
BOD	mg/L	706.78	706.78	409.93	225.46	225.46	4.85	3.33	3.33	30
TSS	mg/L	798.97	798.97	319.59	55.93	55.93	0.63	0.02	0.02	30

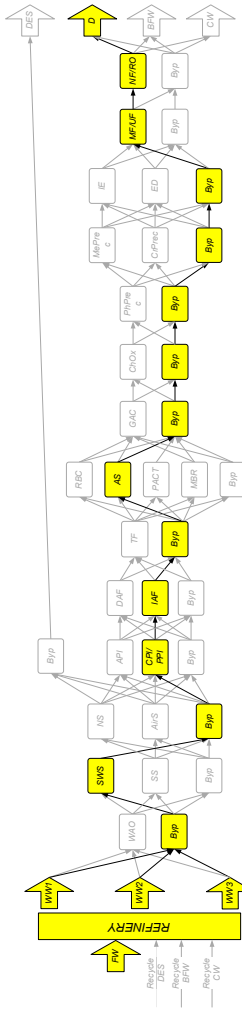


Figure 10.3. WTR solution 1: Step 3 results - Optimal configuration in deterministic conditions

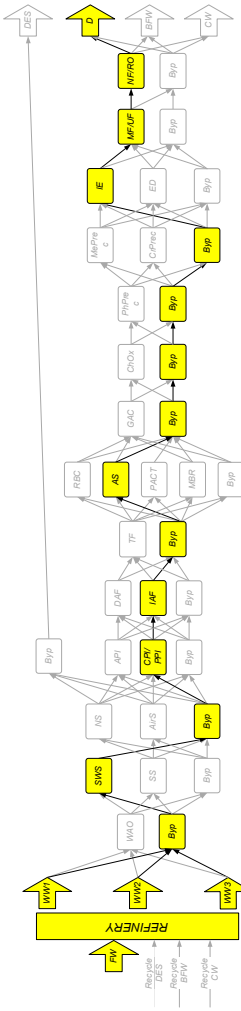


Figure 10.4. WTR solution 1: Step 5 results - Optimal configuration under uncertainty

10.1.5 Step 5: Solution under uncertainty

The definition of the uncertainty with respect to influent composition is incorporated in the design problem, and the problem under uncertainty is formulated, aiming at the identification of a treatment network which is feasible over the entire uncertainty domain and whose expected total annualized cost is minimal. The solution is reported in figure 10.4.
With respect to the solution obtained under deterministic condition, ion exchange is added as tertiary treatment, in order to assure compliance with the regulation for Cr^{6+} emissions.
As highlighted in table 10.7, robustness against wastewater composition uncertainty is obtained at the price of a relevant decrease in financial performances for the treatment, with the total annualized cost increasing by over 36%.
The root cause of such a dramatic performance erosion can be identified in the wide range of variation for wastewater composition defined in step 2, which the treatment plant have to be able to manage. Consequently, the analysis suggests focusing on the reduction of such a data uncertainty through further investigation of wastewater composition.

Table 10.6. Oil refinery wastewater treatment and reuse solution 1 - Uncertainty mapping: results details

		Treatment Steps													
Net.	**Prob.**	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**IX**	**X**	**XI**	**XII**	**XIII**	**XIV**	**Eff.**
1	62.0%	byp	SWS	byp	CPI/PPI	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	NF/RO	D
2	11.5%	byp	SWS	byp	CPI/PPI	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	byp	D
3	9.0%	byp	SWS	byp	CPI/PPI	IAF	byp	AS	byp	byp	byp	byp	byp	MF/UF	NF/RO	D
4	8.5%	byp	SWS	byp	API	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	byp	D
5	4.0%	byp	SWS	byp	API	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	NF/RO	D
6	2.0%	byp	SWS	byp	CPI/PPI	IAF	TF	AS	byp	byp	byp	byp	IE	byp	byp	D
7	1.5%	byp	SWS	byp	API	IAF	TF	AS	byp	byp	byp	byp	IE	MF/UF	byp	D
8	0.5%	byp	SS	byp	CPI/PPI	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	NF/RO	D
9	0.5%	byp	SWS	byp	API	IAF	byp	AS	byp	byp	byp	byp	byp	MF/UF	NF/RO	D
10	0.5%	byp	byp	AirS	CPI/PPI	IAF	byp	AS	byp	byp	byp	byp	IE	MF/UF	byp	D

Table 10.7. Oil refinery wastewater treatment and reuse solution 1 - Comparison of deterministic solution and solution under uncertainty

Indicator		Deterministic	Under uncertainty	% Diff
TAC	**(M$/yr)**	**17.454**	**23.762**	**36.10%**
Capex	(M$)	22.8	25.217	10.60%
Opex	(M$/yr)	15.934	22.081	38.60%
Utility Cost	(M$/yr)	11.049	11.535	4.40%
Waste Cost	(M$/yr)	4.886	10.546	115.90%

10.1.6 Step 7: Results report

The results obtained by the analysis have been aggregated in a report, presented in table 10.8.
The Expected Value of Perfect Information (EVPI) is calculated as 5.541 M$/yr, corresponding to more than 30% of the TAC obtained in deterministic condition. This high EVPI value indicates that a considerable performance improvement can be obtained by reducing the uncertainty associated with the data. Therefore, it suggests performing more detailed analysis with respect to the composition of the wastewater influent, in order to obtain, if possible, a "more certain" knowledge of such data. In case the root cause of such uncertainty is related to the lack of precise measurements of influent wastewater composition, the analysis suggests performing a detailed experimental campaign. If, on the other hand, the uncertainty is due to the fact that the flow composition is subject to variation over time, the results suggest to evaluate options for flow stabilization (e.g. with buffer tanks).
This observation confirms that, as stated in the introduction, the replication of a standard design for different wastewater compositions does not represent an economically convenient alternative, since a considerable performance penalty is paid to design a treatment able to receive wastewater with different contamination levels.
As described in chapter 4, the calculation of the Value of Stochastic Solution (VSS) requires evaluating the performances of the network obtained as deterministic solution (obtained as step 3 result) for the uncertainty defined in step 2, (equation 4.27).
The result of this analysis shows that, when the uncertainty in the wastewater composition is considered, the configuration selected as optimal deterministic network is unable to guarantee robust compliance with the discharge regulation. In particular, as shown in figure 10.5 the probability of discharging a water effluent violating the concentration limit of Cr^{6+} (0.1 mg/L) is over 70%.
Consequently, the estimation of the VSS has not been performed, since this would require including the consequences of the limits violation (e.g. fines, legal costs, etc.) within the problem boundaries. Nevertheless, even though a quantification of the VSS has not been obtained, the value associated to the consideration of data uncertainty has been demonstrated.
Finally, a considerably high value of Uncertainty Price (UP) is estimated (6.308 M$/yr). This value confirms that, when such a high uncertainty is included in the design problem, the performances of the network are subject to a considerable

reduction.

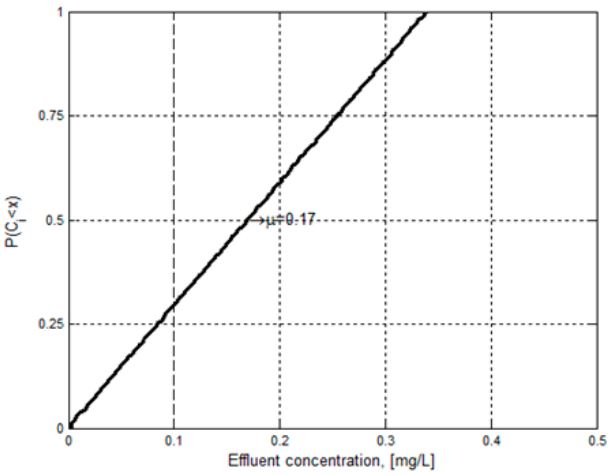


Figure 10.5. WTR solution 1: Cumulative distribution of Cr^{6+} concentration in water effluent. The emission limit for discharge as surface water is 0.1 mg/L

Table 10.8. WTR solution 1: Results report

Solution	Network	TAC (M$/yr)
Deterministic network	byp SWS byp CPI/PPI IAF byp AS byp byp byp byp byp MF/UF NF/RO D	17.454
Network u/ uncertainty	byp SWS byp CPI/PPI IAF byp AS byp byp byp byp IE MF/UF NF/RO D	23.762

Indicator	VSS (M$/yr)	UP (M$/yr)	EVPI (M$/yr)
Network u/ uncertainty	n/a	6.308	5.541

10.2 Oil refinery wastewater treatment and reuse solution 2 (multi-stream)

10.2.1 Step 1: Problem formulation

The formulation of solution 2 is similar to the one described for solution 1, with the exception that, in this case, a multi-stream solution of the deterministic problem is required.
Since data uncertainty is not considered, step 2, 4, 5, 6 and 7 of the workflow are not executed.

10.2.2 Step 3: Deterministic solution

The deterministic formulation results in a problem of the same size as the above described single-stream case (WTR solution 1), but in the form of non-convex MINLP, containing 4,326 bilinear terms. Because of the size of the problem and the number of bilinear terms, direct solution is not possible, and the bi-level decomposition strategy has been used.
In order to facilitate the solution of the problem, the strategies presented in section 5.2.3 have been adopted. The upper bounds for the flow variables have been calculated based on data analysis, according to equations 5.10 - 5.11. The data analysis allowed reducing the number of bilinear terms to 2060 (corresponding to a reduction by 52%), by identifying the variables which can be fixed to zero, and eliminating the corresponding constraints.
Bi-dimensional domain partitioning of the flow variable have been implemented, by dividing each of the variables appearing in the bilinear terms in a 2x2 grid. This resulted in the definition of additional 1806 binary variables. Finally, strengthening cuts derived from mass balance (equation 3.25) and from coherence of flow patterns (equation 5.14-5.15) have been added to the relaxed problem, in order to tighten the relaxation.
The problem has been solved via bi-level decomposition; a solution characterized by a total annualized cost of 13.654 M$/yr, has been identified, with a relative optimality tolerance of 7.5%. The solution required 10 major iterations and a CPU time of 90,238 seconds (25 hours) on a standard computer.
In the treatment configuration obtained (shown in figure 10.6), the incoming wastewater is pretreated by the SWS. The purity of the pretreated water meets the desalter specification, therefore part of the water (7.7% of the total load) is sent back to the refinery bypassing the remaining treatment sections. The rest of water is treated by the CPI/PPI and IAF, followed by AS for biological treatment. Most of the ternary treatments are bypassed, with the exception of Ion Exchange, MF/UF and NF.

10.3 Comparison between single- and multi-stream solution

The comparison of the results for the 2 scenarios (reported in table 10.9 shows that when water recycle is considered, the proposed approach is to identify a win-win

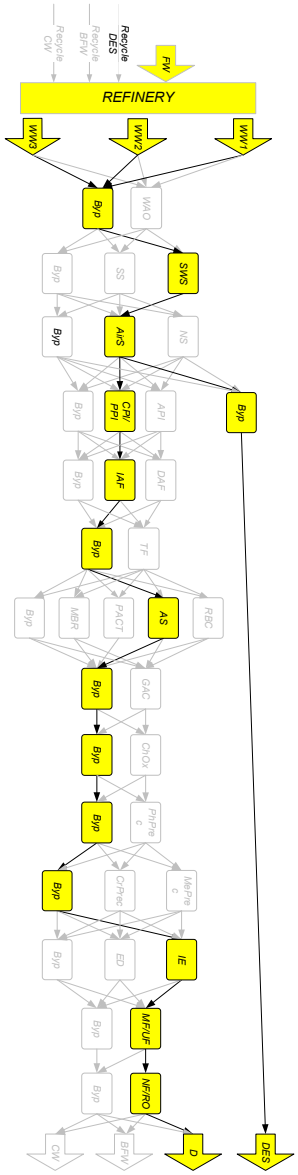


Figure 10.6. Step Results: Water with recycle

solution, meaning a solution in which both the economic objective (Total Annualized Cost) and the sustainability objective (reduction of the water footprint of the refinery) are improved, with respect to the single-stream results. In particular, for this case a reduction of 21.8% of the total annualized cost and of 45.3% of the water footprint is achieved.
Such a solution is identified at the expense of a considerable increase in the numerical complexity of the optimization problem, resulting in a significant increase in computational time required for its solution, as showed in table 10.10.

Table 10.9. Oil refinery wastewater treatment and reuse network: Comparison between single- and multi-stream solution

Item		Solution 1: No Recycle	Solution 2: Recycle	% Diff
Financial indexes				
Total Annualized Cost	**(M$/yr)**	**17.454**	**13.654**	**-21.80%**
Capex	(M$)	22.8	19.859	-12.90%
Opex	(M$/yr)	15.934	12.34	-22.50%
Utility Cost	(M$/yr)	11.049	8.644	-21.80%
Waste Cost	(M$/yr)	4.886	3.696	-24.40%
Savings	(M$/yr)	-	0.074	-
Environmental indexes				
Refinery Water Footprint	**(t/h)**	**208.02**	**113.769**	**-45.30%**
Water Recycled	(t/h)	-	50	-
Water Effluent	(t/h)	449.53	493.735	9.80%
Water Wasted	(t/h)	208.02	113.769	-45.30%
Water Withdrawn	(t/h)	657.504	607.504	-7.60%

The existence of a win-win solution when water recycle is considered is specific to the case study, and cannot be generalized. In this case, the simultaneous improvement of both objectives is due to the reduced load to primary, secondary and tertiary treatments (because of the water recycle to the desalter), which causes a reduction in capital and operational costs for these sections. However, the problem solution strategy is generic and can be applied to explore optimal networks alternatives for different oil refinery wastewater characteristics.

10.4 Conclusions from the wastewater treatment and reuse case study

Through this case study, the capability of the framework to formulate and solve problems related to water management has been demonstrated.
In particular, the framework showed the capability to handle problem of industrially relevant complexity, characterized by a large number of contaminants, flows and treatment intervals, contributing to the effort of bridging the gap between the capabilities of optimization-based design methods and the requirements of wastewater experts and professionals.
In particular, *WTR solution 1* demonstrated the ability of the framework to manage the uncertainty associated with the input data, leading to the identification of

Table 10.10. Oil refinery wastewater treatment and reuse: Solution statistics for single- and multi-stream solution

Problem	Single-stream		Multi-stream
	deterministic	uncertainty	deterministic
Class	MILP	MILP	MINLP
n binary variables	41	41	41
n constraints	51,567	3,632,748	51,567
n non-convex constraints	-	-	2060
Relative optimality tolerance	1.00E-06	1.00E-06	7.50%
CPU time (s)	2.5	460.2	90,238.20
Solution algorithm	direct	direct	bi-level decomp.
n outer iterations	-	-	10
Average CPU time for 1 iteration (s)	-	-	9,023.80

robust solutions.
Moreover, through *WTR solution 2*, the ability of the framework and of the bi-level decomposition strategy to solve the non-convex MINLP problem resulting from the formulation of the multi-stream problem has been demonstrated, even though at the expense of a considerable computational investment.
For a problem of this size and complexity, in fact, over 24 hours of computational time were required in order to obtain the solution. Moreover, the global optimality of such a solution could not be proven with a relative tolerance lower than 7.5%.
The comparison between the performances of the obtained single- and multi-stream solution, though, highlighted that such an increase in complexity (and in computational time) allowed identifying a win-win solution, in which the improvement of both the economical objective and of the water footprint of the refinery is achieved.
The treatment configuration obtained as a result of the computer aided analysis provides design targets for detailed engineering and dimensioning of unit operations involved in the network which can further be simulated and verified with the use of detail models and process simulators (such as WEST) and experimentally verified at pilot scale, if needed, in order to obtain the final design.

Part IV

Conclusions

11

Conclusions and recommendation for future developments

11.1 Summary of the project outcomes

In this Ph.D. project, the problem of synthesis and design of processing networks has been studied, with the aim of simplifying and optimizing its formulation and solution. To this end, a workflow has been developed, in which all tasks related to the formulation and solution of this class of problems are organized. Methods, tools, databases, software and solution strategies supporting the execution of the most important tasks of the workflow have been developed. Finally, all these components have been integrated in a computer-aided framework, designed in order to facilitate the execution of the workflow.

As a result of this project, an integrated business and engineering framework for synthesis and design of processing networks has been developed. Because of its generic structure, the framework can be employed for the formulation and solution of network problems related to different application domains.

Within the framework structure, the formulation and solution of the problem is decomposed into 7 successive steps. The first 2 steps are related to the definition of the problem, and the specification of the domain of uncertainty to consider in the analysis. The solution of the decision-making problem under uncertainty is decomposed through steps 3-6, in which layers of complexity with respect to the consideration of uncertainties are stepwise added in the problem formulation. As a result of this decomposition, a significant amount of insights and information about the consequences of the uncertainties on the decision-making problem is obtained prior to considering the complexity of the entire problem. Consequently, this information is used to facilitate the solution of the problem by for example variable initialization, or employing pragmatic simplification of the optimization under uncertainty problem. In the last step, all results and insights obtained during the execution of the workflow are aggregated in a result report, where all information relevant to decision-making is summarized.

The execution of the workflow is supported by several methods and tools, which have been developed and integrated within the framework. In particular, a multi-scale ontology has been developed in order to represent a wide variety of processes in a generic manner. Through the combination of the different elements of this ontology, all existing alternatives with respect to processing network configuration can be described based on the concept of *process tasks* (at unit operation scale), *intervals* (at process scale) and *steps* (at plant scale), and organized in a superstructure representation.

Based on this ontology, a generic model to describe the elements of the superstructure is developed. Each of the process tasks has been described through a generic process task model. Process interval models are then generated by combining task models. In a similar way, the model of the entire superstructure is obtained by combining different process interval models.
Moreover, to support the task of defining the search space for the optimization problem, a superstructure synthesis method has been developed. This method, based on means-ends analysis, employs engineering and commercial insights to generate the superstructure through a systematic approach.
Furthermore, a multi-layer data structure has been developed, to manage and systematize the data required for the definition of the design problem. The practical implementation of the architecture resulted in the development of EOLO, a problem formulation software integrating automatic data consistency checks and database functionalities. Through EOLO, standard deterministic problems (based on generic process interval models) can be automatically formulated and solved, without requiring any programming in GAMS.
Finally, tailor-made solution strategies based on bi-level decomposition have been developed for the solution of a class of problems, whose direct solution is not convenient or impossible, such as large scale non-linear or non-convex mixed integer problems. These strategies have been integrated through the development of solution files written in GAMS, in which the model reformulation and the dataflow required by these algorithms are partially automated.
The integrated business and engineering framework has been applied through the formulation and solution of 3 case studies, in order to test its functionalities and highlight its features. These case studies, related to different application domains, have been successfully solved for different scenarios, with respect to objective function and data uncertainties definition, as well as with respect to the resulting non-linear and non-convex type optimization formulations as found in multi-stream problems.

11.2 Fulfilment of project objectives

From the results presented in this thesis, it can be seen that the objectives defined for this project (section 1.3) have been fulfilled through the work developed in this Ph.D. project. In particular:

- a workflow for the formulation and solution of the problem of synthesis and design of processing networks has been developed, together with the models, methods, tools and solution strategies needed for its execution.
- a computer-aided framework integrating all the above mentioned components has been developed, and applied to the formulation and solution of 3 case studies from different application domains.

11.3 Scope and significance

Optimization-based design methods based on mathematical programming are based on the decomposition of the design problem in three steps: the definition of the

search space, the formulation of an optimization problem and the identification of the best alternative. These steps represent the practical implementation of rational decision-making, as defined in the seminal work of Simon (1947). When a real problem is concerned, practical limitations arise, reducing the efficacy of these methods. These limitations may be described as (Simon, 1972):

1) the decision-maker may ignore some of the possible alternatives

2) the decision-maker may not have enough time/resources to evaluate all alternatives (because of their number, of the complexity of the evaluation, of time constraints)

3) the decision-maker may ignore the exact value of some of the data needed to evaluate the alternatives

4) different and conflicting goals may exist

From this perspective, it can be seen how the methods and tools developed and integrated in the framework developed in this Ph.D. project have contributed to managing these limitations, hence promoting the conditions for rational design of processing networks. In particular:

- the superstructure synthesis method presented in chapter 6.1 contributes to the formulation of superstructures corresponding to large search spaces for the optimization problem, including innovative solutions.
- the practical implementation of the developed framework contributes to optimize the workflow of formulation and solution of optimization problems, facilitating the comparison of large number of alternative configurations.
- optimization under uncertainty is used to take into account the limited knowledge with respect to some of the problem data, allowing to identify robust solutions.
- the flexibility of the framework with respect to problem formulation allows studying the effect of different objectives and the identification of trade-off solutions, (e.g. between economics and sustainability objectives).

Based on these observations, it can be concluded that the work presented here has contributed to moving beyond the state of the art of optimization-based methods for synthesis and design of processing networks.

11.4 Remaining challenges

While the work presented here contributed to optimizing the workflow of formulation and solution of processing network problems to a significant extent, a number of areas may still benefit from further attention and development.

11.4.1 EOLO

Given the ability that EOLO has shown in simplifying the task of formulating standard problems, its expansion in order to cover the formulation of more classes of problems is highly advisable. To this goal, further developments of EOLO are needed in order to:

- expand the data architecture to include the specification of problem under uncertainty.
- develop the automatic GAMS file generation feature for multi-stream and stochastic problems.

11.4.2 Model library

Although a wide range of problems can be formulated and solved based on the models currently included in the model library, its expansion is required in order to expand further the scope of application of the business and engineering framework. In particular, suggested directions for development include:

- expanding the model library by including economic and sustainability models.
- complementing the task model library with a library of constitutive models, to be used in order to calculate reaction conversions, separation factors, etc. as a function of flow composition and process parameters.

11.4.3 Solution methods

In order for optimization-based design methods to reach their full potential, solution methods able to handle the solution of the large and complex discrete optimization problems that are formulated are required.
Although the solution methods integrated in the framework have shown the potential of solving problems of considerable size and complexity, developments are required for:

- further reducing the time and computational resources needed for the solution of the optimization problems.
- allowing the solution of more complex problems, such as multi-stream stochastic problems, or problems containing non-convex constitutive equations.
- developing a decision tree to guide the selection of the most appropriate solution method, depending on size and structure of the problem to be solved

11.5 Future perspectives

Based on the results obtained in this work, as well as on the feedbacks received while presenting the integrated business and engineering framework in industrial forums and scientific conferences, some perspectives on future development of the concepts presented here have been developed.
In the next sections, these suggested developments will be briefly discussed.

11.5.1 Network-of-networks

The multi-scale structure of the framework could be expanded, in order to include the complexity of the entire enterprise, by representing it as a *network-of-networks*, as described by figure 11.1. Within this representation, the enterprise is modeled as a network of processing networks, each of them constituted by a network of processing plants, constituted by a sequence of one or more processes.
By capturing the interdependencies between scales in an explicit manner, the network-of-networks approach would enable evaluating the impact that modification operated at process level, plant level and business level would have on the overall enterprise, at all different scales. Consequently, the development of the network-of-networks approach would allow the use of the framework to support decision-making with respect to i) process improvement project (at process scale), ii) construction and decommissioning of entire process plants (at plant scale) and iii) establishment, modification or discontinuation of entire businesses (at enterprise scale).
Furthermore, this expansion would allow evaluating the consequences of changes in the business environment (with respect to market conditions, raw material availability and regulations) on the performances of the overall company, rather than on individual businesses.

11.5.2 Virtual technology customer

The ability of the framework to evaluate different alternatives with respect to process technology, in order to identify the most convenient selection could represent a valuable tool also for technology developers and equipment manufacturers (developing and producing process technologies and selling them to processing companies), in order to estimate the value of their products, from a customer perspective.
By formulating a processing network design problem, including their technology among the alternatives of the superstructure, in fact, technology developers could have a tool reproduce the decision-making of their customers. From this perspective, therefore, the integrated framework could be modified to develop a *virtual technology customer* model, which could be used in order to study the competition among different technologies, from a cost/ benefit perspective. By simulating the decision-making of processing companies with respect to technology selection, in fact, the virtual customer model could be used by technology developers to assess the competitive advantage/ disadvantage of their technologies with respect to the competition, and use this information in order to guide their bidding, marketing and development decisions.
Finally, the virtual technology customer model could be used as a tool to estimate the value that a given technology has for its buyer. Such estimation constitutes a valuable information for technology developers, and could be used as input for the definition of the technology price, which could be defined according to the value-share principle (World Intellectual Property Organization, 2004).

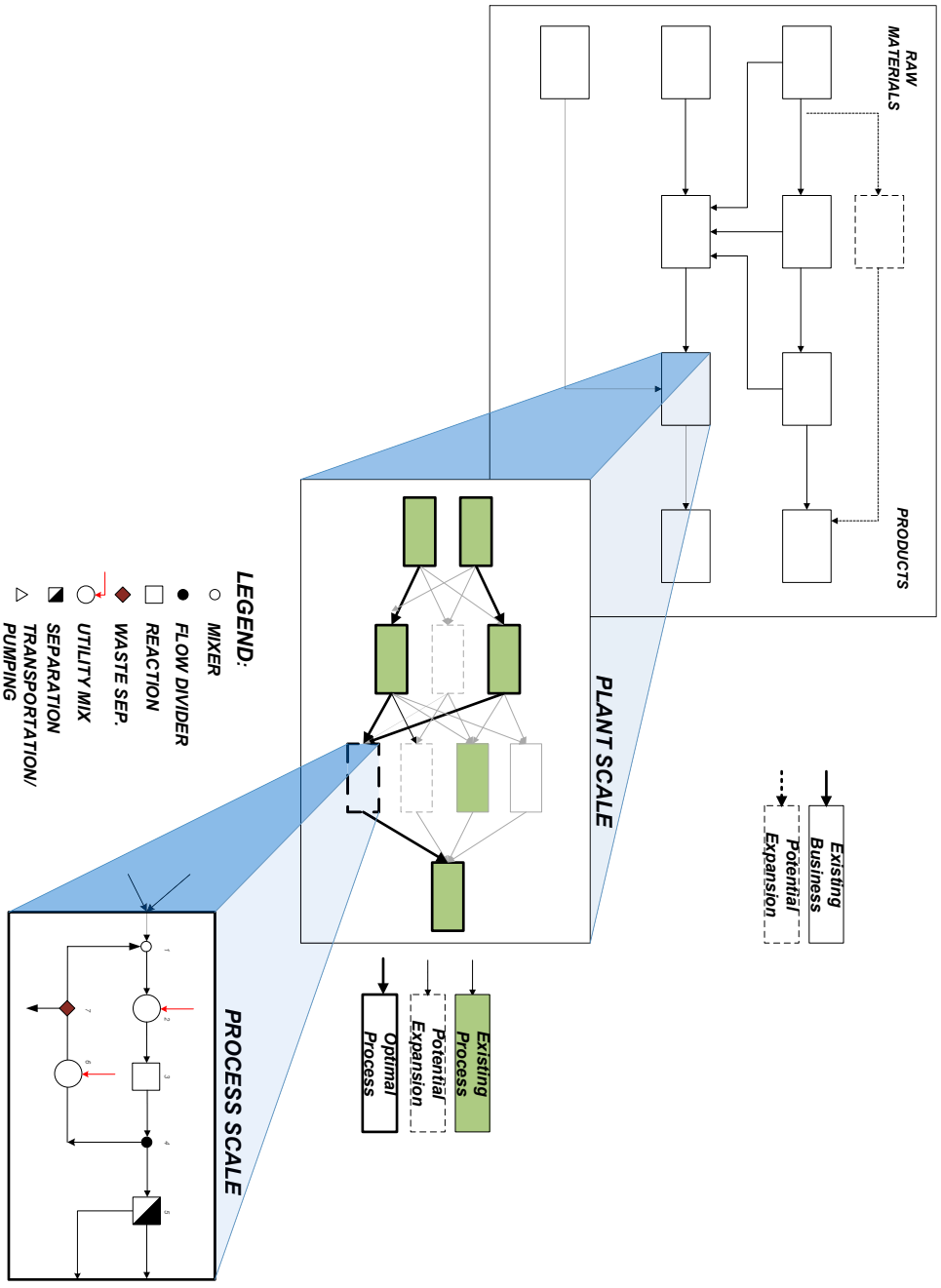


Figure 11.1. Example of the representation of an enterprise as a network-of-networks

11.5.3 Connection to process simulator for detail verification

A connection between the computer-aided framework developed here and a process simulator could be developed, in order to automatically generate a detail model, for the verification of the processing network configuration obtained as result of the optimization problem. Through the development of this connection, a computer-aided tool able to support the entire development funnel (described in section 1.3) could be obtained. This would allow further integration of the workflow for synthesis and design of processing networks, hence reducing the time and resources needed for this task.

Acknowledgement of the financial support

The research leading to these results has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n 238013.

Appendices

A

Nomenclature

Indexes

i	Set of components
j	Set of partitions for piecewise linearization of capital cost
k	Set of process intervals (source)
kk	Set of process intervals (destination)
N_i	Number of components (index k)
N_k	Number of process intervals (index k)
N_L	Number of flow intervals (index j)
N_S	Number of Monte Carlo samples (index s)
N_{ST}	Number of process steps (index st)
o	Set of partitions for McCormick relaxation (first level)
oo	Set of partitions for McCormick relaxation (second level)
p	Set of partitions for McCormick relaxation (first level)
pp	Set of partitions for McCormick relaxation (second level)
$PROD(k)$	Subset of product process intervals
$RAW(k)$	Subset of raw material process intervals
$react(i)$	Subset of key reactant components
s	Set of Monte Carlo samples
st	Set of processing steps
$ut(i)$	Subset of utility components

Problem data

$\alpha^{i,k}$	Fraction of utility i mixed with process stream in interval k
$\gamma^{i,k,rr}$	Stoichiometry of component i in reaction rr occurring in interval k
$\Gamma^{k,j}$	Grid for flow partitioning
$\delta^{i,k}$	Fraction of component i separated as waste in interval k
$\eta^{k,kk}$	Transportation distance (or pressure drop) between interval k and interval kk
Λ^{p}	Grid for split factor partitioning
$\mu^{ut,i,k}$	Specific consumption of utility ut with respect to component i in interval k
$\nu^{k,st}$	Allocation of interval k to step st (binary)
π_R^k	Cost of raw material k
$\pi_U^{i,k}$	Cost of utility i for interval k
π_P^k	Value of product k
$\pi_T^{k,kk}$	Price for transportation between interval k and kk
π_{Ca}^k	Coefficients for capital cost calculation
π_{Cb}^k	Coefficients for capital cost calculation
$\widehat{\pi}_{Ca}^{k,j}$	Capital cost coefficients (piecewise linearized model)
$\widehat{\pi}_{Cb}^{k,j}$	Capital cost coefficients (piecewise linearized model)
$\sigma^{i,k}$	Fraction of component i collected in primary of separator in interval k
τ	Investment time horizon
$\theta^{i,k}$	Conversion of key reactant react in reaction rr in interval k
$\upsilon^{k,st}$	Allocation of intervals to process step (binary)

$\zeta^{k,kk}$	Superstructure connections (binary)
$\zeta_P^{k,kk}$	Primary superstructure connections (binary)
$\zeta_S^{k,kk}$	Secondary superstructure connections (binary)
$\omega^{k,kk}$	Fraction of outlet flow of interval k which is fed to interval kk
MW^i	Molecular weight of component i
dr	Discount rate
Functions	
$E_\theta(f)$	Expected value of function f over the domain of
P_s	Probability of realization of event s
Continuous variables	
$CAPEX$	Capital investment
$EBIT$	Earning Before Interest and Tax
$F_{in}^{i,k}$	Flow of component i entering interval k
$F_M^{i,k}$	Flow of component i after utility mix in interval k
$F_R^{i,k}$	Flow of component i after reaction in interval k
$F_W^{i,k}$	Flow of component i after waste separation in interval k
$F_{out1}^{i,k}$	Flow of component i in primary outlet of separation in interval k
$F_{out2}^{i,k}$	Flow of component i in secondary outlet of separation in interval k
$W^{i,k}$	Waste flow of component i in interval k
$F^{i,k,kk}$	Flow of component i from interval k to interval kk
$F_1^{i,k,kk}$	Flow of component i from interval k to interval kk (primary)
$F_2^{i,k,kk}$	Flow of component i from interval k to interval kk (secondary)
$F_D^{k,j}$	Disaggrageted flow
$F_{D_{out,f}}^{i,k,kk,o,p}$	Disaggregated Flow variable for domain partitioning
GOI	Gross operating income
$GREV$	Gross revenues
INV^k	Investment cost for interval k
NPV	Net Present Value
$OPEX$	Operational investment
$R^{ut,k}$	Flow of utility ut entering interval k
R_{cost}	Raw material cost
$SM_f^{k,kk}$	Fraction of outlet flow f of interval k fed to interval kk
$SM_{D_f}^{i,k,kk,o,p}$	Disaggregated split factor variable for domain partitioning
T_{cost}	Transportation cost
U_{cost}	Utility cost
W_{cost}	Waste disposal cost
Binary variables	
y^k	Selection of process interval k
y_I^k	Selection of process interval k (first stage)
y_{II}^k	Selection of process interval k (second stage)
$v^{k,j}$	equal to 1 if flow of process k is in partition j
$w_f^{i,k,kk,o,p}$	equal to 1 if flow and SM of process k are in partition o,p

Miscellanea	
VSS	Value of Stochastic Solution
UP	Uncertainty Price
$EVPI$	Expected Value of Perfect Information
UB	Upper Bound of the objective function
LB	Lower Bound of the objective function
UP	Variable upper bound
LO	Variable lower bound
Components	**Soy processing case study**
TAG	Triacylglycerides
DAG	Diacylglycerides
MAG	Monoacylglycerides
HP	Hydratable Phosphorous (lecithin)
NHP	Non-Hydratable Phosphorous
Steam-LP	Low pressure steam (3bar)
Steam-MP	Medium pressure steam (10bar)
Steam-HP	High pressure steam (60bar)
Components	**Wastewater treatment and reuse case study**
COD	Chemical Oxygen Demand
BOD	Biological Oxygen Demand
FSS	Fixed Suspended Solids
TSS	Total Suspended Solids
$O\&G$	Oil and Grease
Treatments	Water treatment and reuse case study
WAO	Wet Air Oxidation
SWS	Sour Water Stripper
SS	H_2S Stripper
NS	NH_3 Stripper
AirS	Air Stripper
API	American Petroleum Institute Separator
CPI/PPI	Corrugated and Parallel Plate Separator
DAF	Dissolved Air Flotation
IAF	Induced Air Flotation
TF	Trickling Filter
RBC	Rotary Biological Contactor
AS	Activated Sludge
PACT	Activated Carbon assisted Activated Sludge
MBR	Membrane Bioreactor
GAC	Adsorption on Granular Activated Carbon
PhPrec	Phosphorous precipitation
MePrec	Metals precipitation
CrPrec	Chromium precipitation
IE	Ion Exchange
ED	Electrodialysis
MF/UF	Microfiltration/ Ultrafiltration
NF/RO	Nanofiltration/ Reverse Osmosis

B

Example of derivation of process interval model

In this appendix, the formulation and solution of the generic process interval models is described through an example. First a process will be proposed and presented through a short process description. Then the input data structure required by the generic process interval models will be explained. Finally, the calculation of the generic model data will be calculated from the process data. Through this example, both the workflow needed for model development and the features of the development model will be highlighted.

B.1 Process Description

The example process is showed in figure B.1, and the equipment list is reported in table B.1.

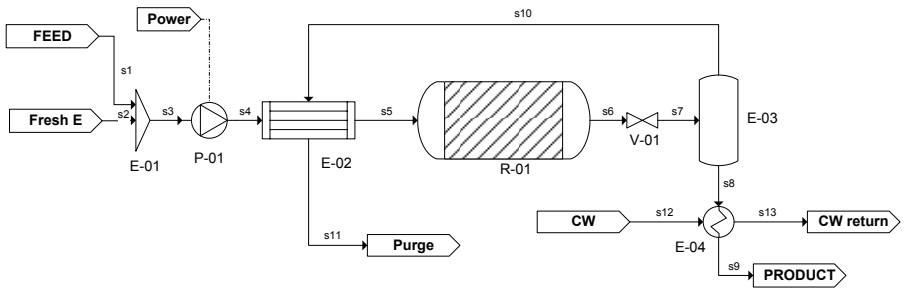


Figure B.1. Process flow diagram for the example process

The feed flow is constituted by a liquid mixture, containing a main component A and impurities B and C. The goal of the process is to reduce the content of component B in the mixture, through a decomposition reaction requiring chemical E:

$$B + 2E \rightleftharpoons C + D \tag{B.1}$$

The reaction is performed in reactor R-01, which constitute the core of the process. The feed flow and the fresh chemical E are fed to mixer E-01.The reaction mixture is than brought to the reaction pressure of 4 bar by pump P-01. In E-02, the mixture is preheated to the reaction temperature of 90°C. The heat duty required by E-02 is entirely provided by the condensation of the recycle stream, and therefore no hot

Table B.1. Example process - equipment list

Tag	Description
E-01	Stream mixer
E-02	Preheater/ condenser
E-03	Flash
E-04	Final Cooler
P-01	Feed Pump
R-01	Reactor
V-01	Release valve

utility is required.
The reaction mixture is fed to reactor R-01, where reaction B.1 takes place. At design condition, 95% conversion with respect to reactant B is achieved. Because of the exothermic reaction, the temperature increases to 235°C.
Through the pressure relief valve V-01, the reactor outlet is fed to the atmospheric flash tank E-03. The flash is adiabatic, and the flash temperature is 150°C. The vapour outlet of flash E-03 is rich in the low boiling point components A and E. In design condition, recovery of A and E is respectively 10% and 100%. The vapour stream S10 is condensed in E-02. Recirculation of the unreacted E is not possible due to regulatory reasons, so stream S11 is sent to blow down.
The liquid outlet of flash E-03 (a mixture of A, B, C and D) is cooled in the final cooler E-04 to the final product temperature of 75°C. A stream table for the process is reported in table B.2.

Table B.2. Example process - stream table

		s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13
tot mass flow	*(kg/m)*	100	22.5	122.5	122.5	122.5	122.5	122.5	90.4	90.4	32.1	32.1	555	555
mass flow A	*(kg/m)*	78	0	78	78	78	78	78	70.2	70.2	7.8	7.8	0	0
mass flow B	*(kg/m)*	15	0	15	15	15	0.8	0.8	0.8	0.8	0	0	0	0
mass flow C	*(kg/m)*	7	0	7	7	7	19.5	19.5	19.5	19.5	0	0	0	0
mass flow D	*(kg/m)*	0	0	0	0	0	12.5	12.5	0	0	12.5	12.5	0	0
mass flow E	*(kg/m)*	0	22.5	22.5	22.5	22.5	11.8	11.8	0	0	11.8	11.8	0	0
mass flow CW	*(kg/m)*	0	0	0	0	0	0	0	0	0	0	0	555	555

Table B.3. Example process - molecular weights

Component		**MW**
A	*(g/mol)*	180
B	*(g/mol)*	48
C	*(g/mol)*	42
D	*(g/mol)*	42
E	*(g/mol)*	18
CW	*(g/mol)*	18

B.2 Generic Process Interval model development

The first step of the process interval model development is the identification of process input-output. The proposed process has 3 material input and 3 material output, plus the power input to the pump. The allocation of these input-output to the category considered in the generic process interval model is reported in Table B.4.

Table B.4. Example process - input output assignment

Inlet		**Outlets**	
Name	**Category**	**Name**	**Category**
FEED	*Process input*	Product	*Process primary outlet*
Fresh E	*Utility input*	Purge	*Waste*
Power	*Utility input*	CW return	*Utility return*
CW	*Utility input*		

Utility and chemical use

Utility and chemical use are described by $\mu^{ut,i,kk}$ and $\alpha^{i,kk}$. The former describes the amount of utility j which is consumed in interval kk as function of the mass flow of component i in the incoming flow. In the proposed process, 3 utilities are consumed.

Fresh E

The flow of fresh E is dosed in order to keep the desired concentration in the inlet of the reactor. As stated in the process description, an excess of 100% of E (over stoichiometry) is required to obtain the desired conversion of B. From the stoichiometry the required flow of fresh E can be calculated as:

$$R^{'E'} = 2 \cdot \frac{\gamma^{'E'}}{\gamma^{'B'}} \cdot \frac{MW^{'E'}}{MW^{'B'}} \cdot F_{in}^{'B'} - F_{in}^{'E'} \tag{B.2}$$

Since $F_{in}^{'E'} = 0$:

$$\mu^{'B','E'} = 2 \cdot \frac{\gamma^{'E'}}{\gamma^{'B'}} \cdot \frac{MW^{'E'}}{MW^{'B'}} = 15 \tag{B.3}$$

since the utility is mixed with the process stream, $\alpha^{'E'} = 1$.

Cooling water

The amount of cooling water consumed in the final cooler is calculated solving the heat balance for the cooler:

$$R^{'CW'} = F_{S8} \cdot c_{p,S8} \cdot \frac{T_{S8} - T_{S9}}{c_{p,CW} \cdot \Delta T_{CW}} \tag{B.4}$$

Where F_S, c_p, S and T_S are respectively mass flow, heat capacity and temperature of stream s, and ΔT_{CW} is the temperature change in cooling water temperature.

From this:

$$\mu^{'CW',i} = F_{S8} \cdot c_{p,S8} \cdot \frac{T_{S8} - T_{S9}}{c_{p,CW} \cdot \Delta T_{CW}} \cdot \frac{1}{F_{IN}} \tag{B.5}$$

Since cooling water is not mixed with the process stream, $\alpha^{'CW'} = 0$

Power

The power input of pump P-01 is calculated as:

$$W_{P-01} = F_{S3} \cdot H_{P-01} \cdot \eta_{P-01} \tag{B.6}$$

Where W_{P-01}, H_{P-01} and η_{P-01} are respectively total power input, total dynamic head and the efficiency of pump P-1, and F_{S3} is total mass flow for stream S3. Consequently:

$$\mu_{power} = H_{P-01} \cdot \eta_{P-01} \tag{B.7}$$

Since power is not mixed with the process stream, $\alpha_{power} = 0$.

A data structure for utility and chemical use is reported in table B.5 - B.6.

Table B.5. Utility consumption data $\mu^{ut,i,k}$ for the example process interval k

$\mu^{ut,i}$	**A**	**B**	**C**	**D**	**E**
E	0.00	15.00	0.00	0.00	0.00
CW	6.14	6.14	6.14	6.14	6.14
Power	0.44	0.44	0.44	0.44	0.44

Table B.6. Utility mixing table $\alpha^{i,k}$ for the example process interval k

α^{i}	
E	1
CW	0
Power	0

Remark:

It is important to underline that the elements of the utility matrix μ can have different measurements units. As an example, data referred to fresh E are in kg/m, while the one relative to power are in $kW \cdot m/kg$.
Depending on the problem, it might be necessary to use different measurement units also for homogeneous data, in order to ensure good problem scaling. Particular care has to be taken in order to ensure dimensional consistence of the data employed in the model.

Reaction

The reaction function is described by use are described by $\gamma^{i,k,rr}$ and $\theta^{react,k,rr}$. The former contains the molar stoichiometric coefficients for component i in reaction

rr. The latter contains the conversion of reaction rr in interval kk, expressed with respect of the reactant react (key or limiting reactant). From equation B.1, the stoichiometry table can easily be obtained (see table B.7). As previously stated, a conversion of 95% of B is required. Component B is therefore selected as key component. The conversion table is reported in table B.8.

Table B.7. Stoichiometry data $\gamma i, k, rr$ for the example process interval k

$\gamma^{i,k,rr}$	**Reaction1**
A	0
B	-1
C	1
D	1
E	-2

Table B.8. Conversion data $\theta^{react,k,rr}$ for the example process interval k

$\theta^{react,k,rr}$	Reaction1
B	0.95

Waste

The separation of wastes from the product stream is modeled via $\delta^{i,k}$, which represents the fraction of component i which is separated in a waste stream. From the process description it can be seen that this corresponds to the component split factors in the flash drum. The resulting waste separation data is reported in table B.9.

Table B.9. Waste separation data $\delta^{i,k}$ for the example process interval k

$\delta^{i,k}$	
A	0.1
B	0
C	0
D	0
E	1

Product separation

Product-product separation is intended as the use of a separation technique on the process flow, resulting in 2 process streams of different composition, named primary and secondary flow. Product-product separation is modeled via $\sigma^{i,k}$, which represents the fraction of component i flow which goes to the primary flow. The example process selected as example does not contain any product-product separation, therefore the elements of $\sigma^{i,k}$ are at their default value (equal to 1, table B.10) meaning that no separation is occurring and all mass flow goes to primary outlet.

Table B.10. Waste separation data $\sigma^{i,k}$ for the example process interval k

$\sigma^{i,k}$	
A	1
B	1
C	1
D	1
E	1

Transportation

Transportation cost is modeled via $\eta^{k,kk}$, which represents the geographical distance between interval k and interval kk. Transportation cost is not considered in the selected example, therefore all the elements of $\eta^{k,kk}$ are equal to the default value of zero.

C

NBP problem data

In this appendix, all data related to the formulation of the NBP problem are reported, in the form of data table.

Table C.1. NBP data: superstructure of primary flows

$\zeta_P^{k,kk}$	**I-1**	**I-2**	**II-1**	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**	**V-1**	**V-2**	**V-3**	**V-4**
I-1	0	0	1	1	1	0	0	0	0	0	0	0
I-2	0	0	1	1	1	0	0	0	0	0	0	0
II-1	0	0	0	0	0	1	1	0	0	0	0	0
II-2	0	0	0	0	0	1	1	0	0	0	0	0
II-3	0	0	0	0	0	0	1	0	0	0	0	0
III-1	0	0	0	0	0	0	0	0	1	0	0	0
III-2	0	0	0	0	0	0	0	0	0	0	1	0
IV-1	0	0	0	0	0	0	0	0	0	0	1	0

Table C.2. NBP data: superstructure of secondary flows

$\zeta_S^{k,kk}$	**I-1**	**I-2**	**II-1**	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**	**V-1**	**V-2**	**V-3**	**V-4**
I-1	0	0	1	1	1	0	0	0	0	0	0	0
I-2	0	0	1	1	1	0	0	0	0	0	0	0
II-1	0	0	0	0	0	1	1	0	0	0	0	0
II-2	0	0	0	0	0	1	1	0	0	0	0	0
II-3	0	0	0	0	0	0	1	0	0	0	0	0
III-1	0	0	0	0	0	0	0	0	1	0	0	0
III-2	0	0	0	0	0	0	0	0	0	0	1	0
IV-1	0	0	0	0	0	0	0	0	0	0	1	0

Table C.3. NBP data: separation split factors

$\sigma^{i,k}$	**I-1**	**I-2**	**II-1**	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**	**V-1**	**V-2**	**V-3**	**V-4**
C-1	1	1	1	1	0.95	0.7	1	1	1	1	1	1
C-2	1	1	1	1	0.99	0.1	1	1	1	1	1	1
C-3	1	1	1	1	0.5	0	1	1	1	1	1	1
C-4	1	1	1	1	0.5	0	1	1	1	1	1	1
U-1	1	1	1	1	0	0	1	1	1	1	1	1

Table C.4. NBP data - utility consumption

$\mu^{ut,i,k}$		**II-1**	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**
U-1	**C-1**	0.4	0	0.5	0.9	0	0
U-1	**C-2**	0.4	1.5	0.5	0	2.5	0
U-1	**C-3**	0.4	0	0.5	0	0	0
U-1	**C-4**	0.4	0	0.5	0	0	0

Table C.5. NBP data - utility mixing

$\alpha^{i,k}$	**U-1**
II-1	0
II-2	1
II-3	0
III-1	0
III-2	0
IV-1	0

Table C.6. NBP data - reaction stoichimetry

$\gamma^{i,k,rr}$		**r1**	**r2**
C-2	**II-2**	-1	
U-1	**II-2**	-2	
C-3	**II-2**	1	
C-4	**II-2**	1	
C-3	**III-2**		-1
C-4	**III-2**		1

Table C.7. NBP data - reaction conversion

$\theta^{react,rr,kk}$		**r1**	**r2**
C-2	**II-2**	0.95	
C-3	**III-2**		0.9

Table C.8. NBP data - waste separation fractions

$\delta^{i,k}$	**II-1**	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**
C-1	0	0.05	0	0	0	0
C-2	0.9	0	0.2	0	0.4	0.8
C-3	0	0	0	0.2	0	0.8
C-4	0	0	0	0	0	0
U-1	0	1	0	0	0	0

Table C.9. NBP data - capital cost and maximum throughput

	II-1	**II-2**	**II-3**	**III-1**	**III-2**	**IV-1**
π^k_{Ca}	12500	7500	8000	17500	8000	5000
π^k_{Cb}	0.55	0.65	0.65	0.66	0.5	0.6
$\sum_i \left(F^{i,k^{UP}}_{IN}\right)$	200	125	200	190	140	150

Table C.10. NBP data - raw material composition

i,kk	**1**	**2**
C-1	78	75
C-2	15	20
C-3	7	5
C-4	0	0
U-1	0	0

Table C.11. NBP data - raw material price

π_R^k	**I-1**	**I-2**
	16.00	18.00

Table C.12. NBP data - product prices

π_P^k	**V-1**	**V-2**	**V-3**	**V-4**
	120	35	70	40

Table C.13. NBP data - utility prices

π_U^i	**U-1**
	5

D

Soy Processing additional results

Table D.1. Soybean processing 1 (deterministic) - Utility table for scenario 1 results

Utility		Process Interval						
		8	**11**	**12**	**20**	**25**	**29**	**31**
Steam-LP	unit-mass		0.67		0.49			
Steam-MP	unit-mass	14.12				0.41	0.18	2.69
Steam-HP	unit-mass					1.05		
Cooling-water	unit-mass	1460.00	42.73	20.60		150.64	1.78	
Process-water	unit-mass		0.48					
Hexane	unit-mass	0.05						
Citric-acid	unit-mass			0.01				
Bleaching-earth	unit-mass				0.21			

Table D.2. Soybean processing 1 (deterministic) - Detail results comparison for the 4 scenarios

PRODUCTS		**scenario 1**	**scenario 2**	**scenario 3**	**scenario 4**
total prod. flow	unit-mass	94.16	93.07	92.72	94.16
FADD	unit-mass	0.47	0.50	0.54	0.47
Refined oil	unit-mass	20.09	19.99	20.05	20.09
Lecithin	unit-mass	0.45	0.45		0.45
LowPro meal	unit-mass	73.15			73.15
HighPro meal	unit-mass		68.86	68.86	
Hulls	unit-mass		3.27	3.27	
PRODUCTS YIELD					
Inlet-to-product	%mass	0.94	0.93	0.93	0.94
FADD	%mass	0.00	0.01	0.01	0.00
Refined oil	%mass	0.20	0.20	0.20	0.20
Lecithin	%mass	4.50E-03	4.50E-03		4.50E-03
LowPro meal	%mass	0.73			0.73
HighPro meal	%mass		0.69	0.69	
Hulls	%mass		0.03	0.03	
UTILITY CONS.					
total utility flow	unit-mass	1696.10	1644.57	1598.02	1696.10
Steam-LP	unit-mass	1.15	1.15	0.49	1.15
Steam-MP	unit-mass	17.40	16.91	16.54	17.40
Steam-HP	unit-mass	1.05	1.04	1.05	1.05
Cooling-water	unit-mass	1675.75	1624.72	1579.68	1675.75
Process-water	unit-mass	0.48	0.48		0.48
Hexane	unit-mass	0.05	0.05	0.05	0.05
Citric-acid	unit-mass	0.01	0.01	0.01	0.01
Bleaching-earth	unit-mass	0.21	0.21	0.21	0.21
WASTE PROD.					
total waste flow	unit-mass	6.99	7.31	7.94	6.99
Exhaust clay	unit-mass	0.26	0.26	0.26	0.26
HP	unit-mass			0.60	
NHP	unit-mass			0.03	
Process-water	unit-mass				
Water	unit-mass	6.30	6.30	6.30	6.30
Fiber	unit-mass		0.32	0.32	
Steam-10b	unit-mass	0.41	0.41	0.41	0.41
Hexane	unit-mass	0.01	0.01	0.01	0.01

E

Oil Refinery Wastewater Treatment and Reuse Technologies

In this appendix, the database of technologies for wastewater treatment is presented, by briefly describing each technology, and highlighting the derivation of the model parameters used for their description in the WTR case study presented in chapter 10.

E.1 Gravity separation

Gravity separators exploit density differences to separate contaminants which are heavier and lighter than water: free oil floats to the surface and solids settle to the bottom, where they are skimmed and scraped off, respectively. While sedimentation is generally employed in all process industries, the following equipments for oil-water separation are typical of refineries.

E.1.1 American Petroleum Institute (API) separator

The American Petroleum Institute (API) separator is a gravity separator which is normally constituted by a long rectangular basin with enough retention time for the oil phase to separate and float to the surface, and laminar flow conditions in order to allow the sedimentation of solids to the bottom.
Usually, scrapers are used to move the oil to the downstream end of the separator where it is collected. On their return to the upstream end, the scrapers travel along the bottom moving the solids to collection. API separators are in general used in the upfront of the treatment train, in order to protect downstream equipments against large oil slugs (European Commission Integrated Pollution Prevention and Control, 2003; Schultz, 2005).
An API separator is effective to remove emulsified particles down to 15 μm of diameters, and typically it is able to reduce oil and total suspended solids down to 50 mg/L and 200 mg/L respectively, for a wide range of incoming inlet conditions (Schultz, 2005).

Removal efficiency

Average removal efficiency is 30.5% for COD and BOD, of 30.0% for TSS and 79.5% for oil and grease (Wong and Hung, 2004; General Electrics, 2012).

Utility consumption

API separators consume electricity to pump fluid through the basin and move the scraper. Correlations of operating costs as a function of the flow rate are reported in Puckett (2008) for a specific API (depth of 1 m, width of 2 m, mean inlet particle diameter of 1 mm). It is reasonable to assume that electricity consumption constitutes the most relevant source of operating cost. Based on this assumption, the specific electricity consumption of an API separator is estimated in μ_{EL,H_2O}=0.22 MJ/ton.

E.1.2 CPI/PPI separator

Corrugated and parallel plate separators (CPI and PPI, respectively) rely on the same principle of the API separators, using tilted plates to increase the collection area while decreasing the overall size of the unit. As the water flows through the separator, the oil droplets coalesce on the underside of the plates and travel upwards to where the oil is collected, while the solids deposit in the bottom. Because of the coalescing action, these separators can separate oil droplets as small as 6 μm in diameter and oil concentrations in the effluent is as low as 10 mg/L (Wong and Hung, 2004).

Removal efficiency Removal efficiencies for CPI or PPI separators for TSS and oil and grease are 60.0% and 90.0% respectively (Zarooni and Elshorbagy, 2006).

Utility consumption CPI/PPI separators consume electricity to pump fluid through the basin and move the scraper. Due to the lack of data specific to this technology, the electricity consumption is assumed to be the same as for an API separator.

E.2 Flotation

Flotation employs chemicals to enhance the separation of very small or light particles, which cannot be effectively separated by gravity separators, thus realizing a sharper separation in a shorter time. The removal is enhanced by the use of chemical additives: inorganic chemicals (aluminum and ferric salts and activated silica) bind particulate matter together and make bubble entrapment easier; polymers can be used to change the liquid-gas and/or solid-liquid interface properties (Tchobanoglous *et al.*, 2003). Through the manipulation of interface properties, pilot-plant studies have shown the ability to reach an outflow oil concentration of less than 10 mg/L, for the treatment of oil refinery wastewater (Wong and Hung, 2004).

E.2.1 Dissolved Air Flotation (DAF)

The Dissolved Air Flotation (DAF) consists of injection of air in water under pressure, followed by pressure release. Typical operating pressures are in the range of 275-350 kPa (Tchobanoglous *et al.*, 2003). The entire flow is held in a retention tank under pressure for several minutes to allow time for the air to dissolve, then

it is admitted to the flotation tank through a pressure-reducing valve where the air comes out of the solution in very fine bubbles, dragging up the suspended material. In large units, a portion of the effluent is recycled, pressurized and semi-saturated with air, then it is mixed with the unpressurized influent before release in the tank; in this way, the energy for pumping is reduced.

Removal efficiency

General Electrics (2012) reports, for DAF, removal efficiencies for COD, O&G, TSS, and BOD of 35.0%, 77.5%, 82.5% and 45.0%, respectively.

Utility consumption

Utilities consumption associated to DAF are:

- Electricity for water pumping.
- Compressed air for injection.
- Chemicals for coagulation and flocculation.

The specific electricity consumption for standard operating conditions is reported in Wang (2010), as $\mu_{EL,H_2O} = 0.59$ MJ/ton.
According to Tchobanoglous *et al.* (2003), the volume of air to the mass of solids ratio (A/S ratio, in [mL/mg$_{TSS}$]) for systems without recycle can be estimated as:

$$\frac{A}{S} = \frac{1.3 s_a (fP - 1)}{S_a} \tag{E.1}$$

where s_a [mL/L] is the air solubility (s_a =18.7 mL/L at 20C), f is the fraction of air dissolved at pressure P (usually 0.5), P [atm] is the operating pressure, S_a [mL/L] is the influent suspended solid. Thus, the volume of air to be introduced for mass of influent water, in [mL/L], is:

$$\frac{A}{S} \cdot S_a = 1.3 s_a (fP - 1) \tag{E.2}$$

The volume of air to be introduced is computed assuming an average pressure of 312.5 kPa and T=20C. The corresponding mass is found assuming perfect gas behavior and a molecular weight of MW_{air}=29 g/mol, in consistent units:

$$\mu_{air,H_2O} = 1.3 s_a (fP - 1) \cdot \frac{P \cdot MW_{air}}{RT} \cdot \frac{1}{\rho_W} \tag{E.3}$$

being R=8.314 J/molK, the compressed air consumption is calculated as $\mu_{Air,H_2O} = 4.74 \cdot 10^{-2}$ kg$_{Air}$/ton With respect to chemicals consumption, according to Parkash (2003) a specific alum consumption μ_{Alum,H_2O}=1.5·10^{-2} kgAlum/ton is required for the treatment of refinery wastewater treatment.

E.2.2 Induced Air Flotation (IAF)

The Induced Air Flotation (IAF) consists of injection of air in water through a revolving impeller. Advantages of IAF with respect to DAF are a more compact size, a lower capital cost, a higher removal of free oil and suspended solids; while disadvantages are higher power consumption and the presence of hydraulic limitations (Tchobanoglous *et al.*, 2003).

Removal efficiency

With the exception of O&G removal, the same performance of DAF are used. For O&G removal, based on what reported in Wang (2010), 90.0% removal is assumed.

Utility consumption

The utility consumption accounts for:

- Electricity necessary for impeller rotation.
- Compressed air.

Data on global consumption or electricity needed to operate the whole equipment are reported in Wang (2010) as $\mu_{EL,H_2O} = 0.66$ MJ/kg.
Due to lack of data, the air consumption of an IAF unit is assumed equal to the one of the DAF unit.

E.3 Biological treatment

Soluble and suspended organic substances are removed by biological treatment, through the activity of microorganisms operating under aerobic or anoxic conditions. In addition, biological nitrification and denitrification and biological phosphorous removal can also obtained for certain operating conditions.
In these treatments, microorganisms utilize the contaminants present in the wastewater as sources of C, N and P, for production of energy and synthesis of new cellular material and nutrients. Typically, industrial wastewaters contains low concentration of N and P with respect to municipal, and consequently these nutrients may need to be complemented in order to maintain the operation, and avoid incomplete treatment and poor sludge settling and flocculation (Henze *et al.*, 1993a). In aerobic processes, oxygen is required for respiration and it can be supplied by means of an aeration system, usually as air.
While most organic compounds can be degraded biologically, for some of them extremely low degradation rates may be observed, and unique environmental conditions may be required (e.g. pH). Furthermore, complete biodegradation of toxic compounds to CO_2 and H_2O or CH_4 may not always be possible, and biotransformations to different organic compounds can occur (Tchobanoglous *et al.*, 2003)
The substrate to be degraded is quantified through COD. However, only part of the COD is biodegradable (bCOD) as described in chapter 6. Additionally, part of this biodegradable material is in the form of particulate, thus hydrolysis is required prior to its use.

While all these factors need to be considered in a detailed modeling of the biological process, in the generic representation employed in the framework all these informations are considered through the definition of pseudo-components. Detailed models can be found for each process in Tchobanoglous *et al.* (2003).
The production of energy and synthesis of new cellular material are considered together through a definition of a biological pseudo-reaction, which is responsible of carbonaceous removal (Tchobanoglous *et al.*, 2003; Eckenfelder *et al.*, 2009):

$$C + O_2 + Nutrients \rightarrow MO + CO_2 + H_2O + Byproduct \tag{E.4}$$

where C is the carbonaceous substrate and MO is for the biomass produced. When an average COD and biomass composition are used (C as $C_{18}H_{19}O_9N$ and MO as $C_5H_7NO_2$) and byproducts are neglected, the stoichiometric equation becomes:

$$C_{18}H_{19}O_9N + \frac{35-10c}{2}O_2 + (c-1)NH_3 \rightarrow cC_5H_7NO_2 + (18-5c)CO_2 + (8-2c)H_2O \tag{E.5}$$

where only NH_3 has been considered as a nutrient in the stoichiometry. The parameter c is:

$$c = y \cdot \frac{MW_C}{MW_{MO}} \tag{E.6}$$

where y is the mass yield of substrate to biomass, MW_C=393 g/mol and MW_{MO}=113 g/mol. The value of y varies with the process conditions, and for aerobic growth of microbial biomass, can be assumed $y = 0.5$ (Henze *et al.*, 1993a). Considering that in aqueous solution ammonia is considered completely ionized, reaction E.5 becomes:

$$C_{18}H_{19}O_9N + 8.80O_2 + 0.74NH_4^+ \rightarrow 0.50C_5H_7NO_2 + 9.39CO_2 + 4.52H_2O + 0.74H^+ \tag{E.7}$$

where the stoichiometric coefficients have been rounded to the second decimal.
With respect to pH, the reaction occurs in neutral conditions, and a pH range from 6.0 to 9.0 is tolerable.
One of the common challenges in biological treatment processes is to maintain a sufficient nutrients (N and P) concentration, with respect to biodegradable soluble COD to be treated (Tchobanoglous *et al.*, 2003). Nutrients are provided to the system as ammonia for N and as phosphoric acid for P (Eckenfelder *et al.*, 2009).
Finally, one of the challenges of biological treatment of industrial wastewater is that some of the contaminants may be inhibitory or toxic to the microbes (Eckenfelder *et al.*, 2009).
The utilities necessary for biological treatment processes are:

- Electricity for pumping and mixing
- Air supply
  As described above, air is needed as utility for maintaining a minimum dissolved oxygen concentration in the aeration tank. In the model used, the

oxygen needed is computed as the oxygen requirement from reaction E.5 with the relevant conversion and a transfer efficiency of η_{O_2}=15% is assumed, thus the effective amount of oxygen to be fed is:

$$\mu_{O_2,C} = \frac{1}{\eta_{O_2}} \cdot \nu_{O_2,E.5} \cdot \theta_{C,E.5} \cdot \frac{MW_{O_2}}{MW_C} \tag{E.8}$$

From the calculated amount of oxygen, the amount of air is obtained.

- Nutrients
  As described, ammonia is added as a source of nitrogen only if not present in enough quantity in the wastewater with respect to the stoichiometric amount. Only the difference needed to perform the reaction is fed. For ammonia, the transfer efficiency is unitary since it is assumed that all the ammonia added is dissolved into the water. Thus:

$$\mu_{NH_3,C} = \nu_{NH_4^+,E.5} \cdot \theta_{C,E.5} \cdot \frac{MW_{NH_4^+}}{MW_C} \tag{E.9}$$

  From microbiological study, the nutrient ratio should be maintained as N:P=5:1 (Henze *et al.*, 1993a; Eckenfelder *et al.*, 2009). Thus, the amount of phosphoric acid to add is:

$$\mu_{H_3PO_4,C} = \frac{1}{5} \cdot \mu_{NH_3} \cdot \frac{MW_N}{MW_{NH_3}} \cdot \frac{MW_{H_3PO_4}}{MW_P} \tag{E.10}$$

Other utilities specific to each biological treatment are discussed for each particular process in the next sections.
With respect to wastes, these are constituted by the sidestream containing the bacteria formed (e.g. the sludge), as well as the solid and the oil and grease separated. Removal efficiencies of the carbonaceous material in the biological units is quantified through the conversion of reaction E.5, $\theta_{C,E.5}$; the value assigned to it is discussed for each unit, as well as the waste fraction of the other components removed.
In the following sections, the biological treatments process technologies which have been included in the superstructure for oil refinery wastewater treatment are briefly discussed.

E.3.1 Activated sludge

Activated sludge (AS) treatment processes consist of two elements: an aeration and reaction tank, where the conversion is performed, and a settling tank, which separates the biomass from the purified water. Then, part of the biomass is recycled in order to keep the desired concentration of microorganisms and the excess biomass is wasted along with part of the suspended solids present in the influent. Within this general concept, different process configuration may exist. Typical performances for Activated Sludge are outlet concentration lower than 20 mg/L for total suspended solids and lower than 25 mg/L for BOD (Eckenfelder *et al.*, 2009).

Parameters of interest in an activated sludge system are:

- Mixed Liquor Suspended Solids (MLSS) and Mixed Liquor Volatile Suspended Solids (MLVSS):
  Concentration of the solids of recycled sludge.

- Solid Retention Time (SRT):
  Average retention time for sludge

- Hydraulic Retention Time (HRT):
  Average retention time for water

- Volumetric organic loading rate:
  Amount of COD and BOD per aeration tank volume per day.

- Food to microorganism ratio (F/M ratio):
  Ratio between the mass of substrate (usually BOD) fed per day and the mass of biomass.

While these parameters are relevant to the detailed design of the activated sludge unit, they are not considered within the scope of this study. Within the case study, AS is represented by lumping the aeration and the clarification part of the process within a single process interval, as described in figure E.1.

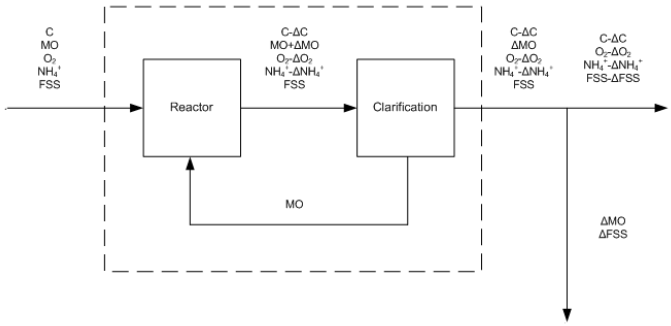


Figure E.1. Representation of activated sludge as a process interval.

Removal efficiency

The conversion for reaction E.5 is assumed $\theta_{C,E.5} = 0.725$ on the basis of the removal efficiency of COD reported in General Electrics (2012). This corresponds to a conservative assumption, since other authors have reported higher efficiencies. For example, Merlo *et al.* (2011) report a conversion up to 93% in a pilot activated sludge for refinery wastewater. On the basis of what reported in General Electrics (2012), the removal efficiency for BOD is 89.5%, for TSS is 72.5%, and for O&G is 89.5%.

Utility consumption

Required utilities are:

- Electricity necessary for the aeration equipment and for the solid recycle.
- Air to supply the necessary amount of oxygen.
- Nutrients necessary to maintain the proper environment for microorganism growth.

The electricity consumption of the activated sludge unit is estimated as an average of the power consumption reported in Environmental Protection Agency (1976) as 2.02 MJ/ton.
The air and the nutrients necessary to operate the system are computed as described for the generic biological unit, with equations E.8, E.9 and E.10 respectively.

E.3.2 Trickling filters

The trickling filters are constituted of a packed bed of rock or plastic covered with biofilm. The wastewater is sprinkled onto the medium through a rotating distribution system above the bed, and flows through the bed.
The trickling filter is followed by a clarifier to settle sloughed-off biofilm and recycle flow may be taken either before or after clarification (Wong and Hung, 2004; Tchobanoglous *et al.*, 2003).
Trickling filters are often used as roughing devices before the wastewater is fed to an activated sludge system, especially in refineries. Typical BOD and total suspended solids effluent concentrations span in the range between 15 and 30 mg/L (Tchobanoglous *et al.*, 2003).

Removal efficiency

Depending on the operating conditions and on wastewater quality, different conversion values are reported for reaction E.5. A typical value of $\theta_{C,E.5} = 0.60$ is used for this study, based on what reported by Qasim (1985) and General Electrics (2012). O&G ans RSS removal of 65.0% and 72.5% respectively are used (General Electrics, 2012).

Utility consumption

The utility consumption accounts for:

- Electricity necessary for operation.
- Air to supply the necessary amount of oxygen.
- Nutrients necessary to maintain the proper environment for microorganism growth.

From the average values reported in Environmental Protection Agency (1976) a specific electricity consumption of 0.51 MJ/ton is calculated. This value confirms how roughing filters require less energy for the treatment of high strength wastewaters as compared to the activated sludge (Tchobanoglous *et al.*, 2003).
The air and the nutrients consumption are calculated based on equations E.8, E.9, E.10 respectively.

E.3.3 Rotating Biological Contactor (RBC)

Rotating biological contactors (RBC) generally consist of rows of polymeric discs mounted on horizontal shafts. The shaft is slowly (1.0 to 1.6 rpm) rotating above a shallow tank, keeping the disc about 40% immersed in wastewater. A biofilm grows on each disk. Aeration occurs while a section of disc is above water level.
Typically, several RBC modules are operated in series. The number of units depends on the treatment goals, and typically four stages are required for BOD removal, while six or more are necessary for nitrification (Wong and Hung, 2004; Tchobanoglous *et al.*, 2003).
RBCs designed for BOD removal only have hydraulic retention time between 0.7 and 1.5 hrs and present an effluent with BOD in the range 15-30 mg/L (Tchobanoglous *et al.*, 2003). Treatment efficiency can be improved by increasing the number of RBCs in series, and by temperature control, sludge recycle, and chemical addition. Their high biomass concentration makes RBCs able to sustain shock loads because of high microorganism concentrations. Moreover, the modular design facilitate their expansion, making them particularly attractive for industrial applications. Full-scale RBC installations in refineries have performances in removal of oxygen-demanding pollutants comparable to activated sludge systems (Wong and Hung, 2004).

Removal efficiency

As for other biological treatment different removal efficiencies are reported in the literature. In particular, with respect to COD removal. The conservative value $\theta_{C,E.5} = 0.80$ is assumed for this study, based on what reported by Tran and Chowdhury (1992). The same source provides an estimation of the removal of BOD and TSS, which for the same reason are chosen as 80.0%.

Utility consumption

The utility consumption accounts for:

- Electricity necessary for operation.
- Air to supply the necessary amount of oxygen.
- Nutrients necessary to maintain the proper environment for microorganism growth.

The electricity consumption of a rotating biological contactor is estimated as an average of the power consumption reported in Environmental Protection Agency (1976), as 2.37 MJ/ton.

The aeration and nutrients supply are calculated based on equations E.8, E.9 and E.10.

E.3.4 Powdered Activated Carbon Treatment (PACT)

The powdered activated carbon treatment (PACT) consists of the use of activated carbon to enhance the performances of the activated sludge. Activated carbon, as described in this appendix, can be used to adsorb wastewater contaminants. By combining the biological and physical action in a single process step, this system is able to buffer toxic loads which might otherwise impair a straight biological system and reduce the amount of carbon otherwise needed by a straight adsorption treatment system.
Consequently, PACT systems allow controlling the removal performances by manipulating the carbon dose. Furthermore, the activated carbon protects the biomass from toxic metals, avoiding the need of up-stream metal precipitation. Finally, the use of carbon leads to a more concentrated sludge (50% solid vs $\leq$20%) with respect to AS (Meidl, 1997).
Depending on the performances required, carbon dosages may vary from 20 to 200 mg/L (Eckenfelder *et al.*, 2009; Tchobanoglous *et al.*, 2003).
Due to the buffering capacity of activated carbon, high contaminant loads can be tolerated (Meidl, 1997).
When there is a small or intermittent application of PAC, the carbon is disposed of with the excess sludge. Continuous application at larger plants, however, requires regeneration of the carbon

Removal efficiency

The COD removal efficiency is obtained as conversion of the growth reaction E.5. A value of $\theta_{C,E.5} = 0.948$ is reported in Meidl (1997). The same source provides an estimation of the removal of BOD and TSS as 99.0% and 92.0%, respectively. A 90.0% removal of O& G is also obtained from Eckenfelder *et al.* (2009).

Utility consumption

PACT requires utility for:

- Electricity for waster pumping.
- Air to supply the necessary amount of oxygen.
- Nutrients necessary to maintain the proper environment for microorganism growth.
- Powdered activated carbon (PAC).

The electricity associated to the PAC feed is neglected, so PACT power consumption is assumed to match the value calculated for AS.
A dosage of 0.11 kg of carbon per ton of water is assumed as PAC consumption.

E.3.5 Membrane bioreactor (MBR)

The membrane bioreactor (MBR) has the same configuration as an activated sludge unit, but membrane separation is used for sludge separation, rather than sedimentation. As a result, higher biomass concentration is achieved in the reactor, hence reducing the reactor volume. Two different MBR process configurations have been applied: external, in which a stream is taken from the reactor and pumped through the membrane unit for the separation, and submerged where the membranes are immersed in the reactor tank. Most submerged membranes are applied under aerobic conditions where aeration is also used to create cross-flow along the membrane to reduce fouling In this study, submerged membranes are considered. Aeration is used both for scouring the immersed membrane and for mixing.

Removal efficiency

A value $\theta_{C,E.5} = 0.90$ is chosen for this technology, to reflect the effect of the membrane separation. The same removal is assumed for the BOD, while for the O&G and TSS removals the same value chosen for the microfiltration unit is used.

Utility consumption

The utility consumption accounts for:

- Electricity for pumping and membrane cleaning.
- Air to supply the necessary amount of oxygen and clean the membrane.
- Nutrients necessary to maintain the proper environment for microorganism growth.
- Chemicals for membrane cleaning.

For anaerobic treatments, submerged membranes (the one considered in this study) are associated to a lower energy consumption (Tchobanoglous *et al.*, 2003), and typical value are 0.92 kWh/m^3 (Judd and Judd, 2006), corresponding to 3.31 MJ/ton. In addition to the air required for oxygen supply for the biological reaction (calculated from E.8), air is also needed for membrane scouring. In many cases the suppliers recommend a proper aeration rate in terms of the specific aeration demand (SAD), either with respect to the membrane area (SADm) or the permeate volume (SADp). Values of SADp range from 10 to 100 Nm^3 of air per m^3 of permeate. When an average value of 55 Nm^3 of air per m^3 of permeate and a recovery of water of 14.5% are assumed, a consumption of 53.84 kgAir/ton is calculated.
Furthermore, backwashing of the membrane is performed with NaOCl. A typical cleaning protocol is based on a weekly cleaning in place (CIP) with 500 mg/L $NaOCl$ and 2000 mg/L citric acid ($C_6H_8O_7$), plus biannual cleaning out of place (COP) using 1000 mg/L NaOCl and 2000 mg/L citric acid Judd and Judd (2006). On this basis, the consumptions are: μ_{NaOCl,H_2O}=0.30·10^{-2} kg$NaOCl$/ton and $\mu_{C_6H_8O_7,H_2O}$=1.24·10^{-2} kg$C_6H_8O_7$/ton.

E.4 Adsorption

Adsorption is in general used to remove those contaminants, which are difficult or impossible to treat by conventional biological treatments, such as, for oil refinery wastewater, heterocyclic organics.
In most of the cases, activated carbon is used as adsorbent, because of its ability to remove a broad range of adsorbates, including many synthetic organic chemicals and inorganics such as heavy metals. Depending on the shape, two types of carbon are usually employed: powdered activated carbon (PAC, 200-400 mesh) and granular activated carbon (GAC, 10-40 mesh). The treatment set-up is based on a column filled with adsorbent, through which the wastwater flows. Two or more columns are usually put in series and rotated as they become exhausted so that the unit remains in operation while one column is taken out of service for regeneration or maintenance (Tchobanoglous *et al.*, 2003).
Generally, the amount of material adsorbed is determined as a function of the concentration at a constant temperature, through the adsorption isotherm. Freundlich is the most used to describe the adsorption characteristics of the activated carbon used in wastewater treatment. According to the literature, typical outlet concentration for adsorption on activated carbon are less than 1 mg/L for TSS, 2 mg/L for BOD and 10 mg/L for COD (Eckenfelder *et al.*, 2009).
The regeneration of the spent carbon can be done via thermal regeneration, steam, or solvent extraction, acid or base treatment, or chemical oxidation (Eckenfelder *et al.*, 2009). Because of the large number of pollutants, thermal regeneration is in general the most convenient regeneration method for wastewater treatment processes (Perry and Green, 2008). Thermal regeneration is the process of drying, desorption, and high temperature heat treatment (650 to 980C) in the presence of steam, flue gas, and oxygen. Usually, for each regeneration cycle 5 to 10 percent by weight of the carbon regenerated is lost due to attrition and burning (Eckenfelder *et al.*, 2009).

E.4.1 Granulated Activated Carbon (GAC)

Typically, a GAC bed treats flow rates between 50 and 400 m^3/h and has cross sectional area between 5 and 30 m^2 and length between 1.8 and 4 m, with a void fraction between 0.38 and 0.42; the approach velocity is 5-15 m/h and the operation time between 100 and 600 days (Tchobanoglous *et al.*, 2003). GAC is normally operated at a wastewater residence time of between 5 and 30 min (Tchobanoglous *et al.*, 2003). For treatment of tertiary effluent, regeneration is economically convenient for capacity higher than 3.0 mgd (0.47 kg/h) (Environmental Protection Agency, 1976). Consequently, activated carbon regeneration has been considered in the case study.

Removal efficiency

COD, BOD, TSS, O&G, and ammonia removal efficiency are estimated as 80.0%, 83.0%, 75.0% and 85.0%, respectively (General Electrics, 2012).

Utility consumption

Utilities include:

- Electricity.
- Make-up carbon.
- Steam, flue gas rich in CO_2 and natural gas for regeneration.

The power consumption is estimated based on the data reported by Environmental Protection Agency (1976) as 3.72 kJ/kg.

Carbon exhaustion rates for refinery wastewater treatment are known to be in the range of 1.1 to 141 lb/1000gal (Eckenfelder *et al.*, 2009), which equals to 0.13 to 16.92 kgGAC/ton. This corresponds to the amount that is treated in the regeneration furnace. Assuming that an average of 7.5% of this carbon is lost, the make-up carbon is μ_{GAC,H_2O}=0.64 kgGAC/ton.
In a municipal wastewater treatment plant, the GAC regeneration uses from 0.259 to 0.375 kg of steam and from 0.475 to 0.687 kgCO_2 per kg of GAC regenerated (Sebastiani *et al.*, 1994). Since specific values for refinery wastewater could not be found, these ranges have been used, resulting in μ_{LPS,H_2O}=12.70 $\cdot 10^3$ GJ/ton and μ_{CO_2,H_2O}=4.96 kgCO_2/ton. The regeneration heat required is 7,000 Btu per lb of carbon regenerated (Environmental Protection Agency, 1976) (16,282 kJ/kgGAC). Natural gas is used as fuel for the furnace, and a specific consumption μ_{NG,H_2O}=2.27 kgNG/ton have been calculated.

E.5 Stripping

Stripping is used in order to remove wastewater contaminants by transferring them to a gas phase. In refinery wastewater, steam stripping is adopted for the separation of H_2S and NH_3 (or NH_4^+), while air stripping is used to remove volatile organic compounds.
In general, strippers involve a downward flow of water through a tray or packed column, while an ascending flow of stripping steam or gas removes the pollutants. The stripping medium can be steam or any inert gas.

E.5.1 Steam stripping

For oil refinery wastewater, 3 steam stripping processes are considered:

1. Two combined columns for stripping of both hydrogen sulfide and ammonia
2. Hydrogen sulfide stripping
3. Ammonia stripping

E.5.1.1 Steam stripping in two columns

In the the majority of the installations in oil refinery wastewater, steam is employed both as heating medium and as stripping gas (Wong and Hung, 2004). Since H_2S tends to be stripped by water more easily than NH_3, if both components need to be stripped, the removal efficiency is enhanced using two columns: in the first, NH_3 is fixed with mineral acid or flue gas and H_2S is stripped, then the ascending steam is condensed and the gas is sent to a sulfur recovery plant; in the second, NH_3 is stripped and recovered as high purity ammonia. The first column is operated at around 40C and the second at around 110C (Eckenfelder *et al.*, 2009).
As a result, an effluent water containing less than 5 mg/L H_2S and 50 mg/L NH_3 can be obtained(Eckenfelder *et al.*, 2009; Wong and Hung, 2004).

Removal efficiency

H_2S and NH_3 removal efficiencies as high as 0.98 and 0.82 respectively have been obtained in single column steam stripping of refinery wastewater Eckenfelder *et al.* (2009). For 2 columns set-ups, H_2S removal is assumed to be 0.98.

Utility consumption

The utility consumption accounts for:

- Electricity necessary to operate the unit.
- Steam as a stripping medium and for temperature regulation.
- Cooling water for temperature regulation.

It is reported that a stripping column for both H_2S and NH_3 has a power consumption of 9.83 MJ/ton Parkash (2003), so it is assumed that two columns would consume 19.66 MJ/ton. From the mass balance:

$$LC_{i,0} + Gy_{i,0} = LC_{i,e} + Gy_{i,e} \tag{E.11}$$

where L and G are the liquid and gas flow respectively (assumed constant), C and y indicate the concentration of pollutant i in the liquid and gas phase respectively, and the subscripts 0 and e indicate the inlet and outlet respectively. Assuming equilibrium between the 2 phases:

$$y_{i,e} = \frac{H_i(T)}{P} C_{i,0} \tag{E.12}$$

where $H_i(T)$ is the Henry constant of component i at the temperature of the column, calculated based on the values reported in Table E.1.
The minimum steam-water ratio is calculated based on thermodynamic equilibrium as:

$$(\frac{G}{L})_{min} = \frac{P}{H_i}(1 - \frac{C_{i,e}}{C_{i,0}}) = \frac{P}{H_i} \cdot r_i \tag{E.13}$$

where r_i is the removal of component i. Based on this minimum value, the operating ratio is obtained as:

$$(\frac{G}{L}) = 2.2(\frac{G}{L})_{min} \tag{E.14}$$

For the removal efficiencies specified above, this results in 0.003 kgLPS/kg for H_2S removal and 0.102 kgLPS/kg for NH_3 removal, for a total of 0.105 kgLPS/kg.
Assuming steam available at 500 kPa and 150 C, the calculated steam consumption corresponds to 218.32 MJ/ton, which falls within the range of typical values reported in the scientific literature for this configuration. According to Eckenfelder *et al.* (2009), in fact, typical steam consumption for the described configuration span between 91.65 and 366.59 MJ/ton.
Besides being used as stripping gas, steam is also employed in order to pre-heat the wastewater to the operating temperature. In order to limit the energy consumption, the energy integration scheme described in figure E.2 is employed. Based on this configuration, the consumption of low pressure steam for pre-heating can be calculated from the energy balance as:

$$\mu_{LPS,H_2O} = \frac{c_{P,w} \cdot (T_{out} - T_{in})}{\Delta H_{vap}} \tag{E.15}$$

where $q = c_{P,w} \cdot (T_{out} - T_{in})$ represents the specific heat given, ΔH_{vap} is the latent heat of the steam, and the other symbols have the same meaning as above.
Similarly, the consumption of cooling water is:

$$\mu_{CW,H_2O} = \frac{c_{P,w} \cdot (T_{in} - T_{out})}{c_{P,w} \cdot \Delta T} \tag{E.16}$$

where $q = c_{P,w} \cdot (T_{in} - T_{out})$ represents the specific heat subtracted to the wastewater to take it from T_{in} to T_{out}, $c_{P,w} = 4.186$ kJ/(K·kg) is the specific heat of water, and ΔT is the temperature variation of cooling water, assumed to be 20C.
From this relations, for the configuration reported in figure E.2 the specific consumption of low pressure steam for heating results in 146.51 MJ/ton, making the total steam consumption equal to 364.74 MJ/ton. However, it is assumed that due to further integration for example with the condensers (in which steam is cooled to recover H_2S and NH_3), this amount is one third lower, thus μ_{LPS,H_2O}=243.16 MJ/ton.
Similarly, the amount of cooling water is instead calculated as μ_{CW,H_2O}=1.75 kgCW/kg. This value appears to be similar to what reported in Parkash (2003), from which a consumption of 1.02 kgCW/kg can be estimated for a 2 column configuration.

Table E.1. Henry constants at 293 K and coefficients to calculate them at various temperatures (Tchobanoglous *et al.*, 2003).

i	$H_i(293K)$ [atm]	A_i [1]	B_i [1]
H_2S	483	884.94	5.703
NH_3	0.75	1887.12	6.315

[1] $H_i(T) = 10^{(-\frac{A_i}{T}+B_i)}$, with T in K

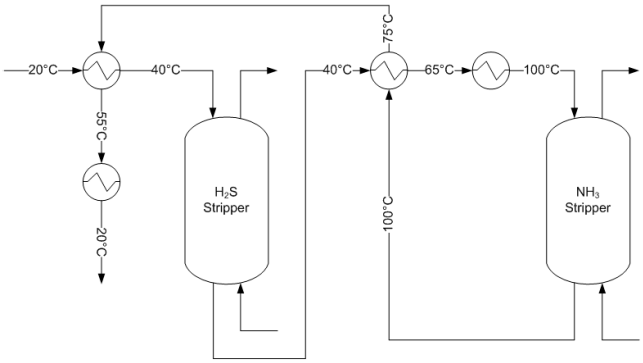


Figure E.2. Scheme of the sour water stripper.

E.5.1.2 Steam stripping of hydrogen sulfide

The stripping steam consumption is only the one for hydrogen sulfide. The cooling water for cooling and the steam for heating are computed with equations E.16 and E.15 following the integration scheme in Figure E.3, as μ_{CW,H_2O}=1.00 kgCW/kg and μ_{LPS,H_2O}=83.72 MJ/ton, respectively. Thus, the total amount of steam is μ_{LPS,H_2O}=89.81 MJ/ton.

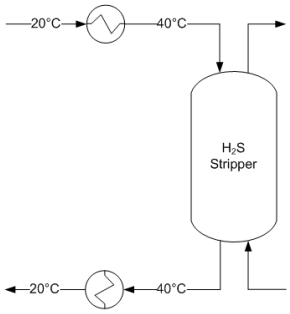


Figure E.3. Scheme of the hydrogen sulfide stripper.

E.5.1.3 Steam stripping of ammonia

For this configuration, only the steam stripping for ammonia stripping is considered. The cooling water for cooling and the steam for heating are computed with equations E.16 and E.15 following the integration scheme in Figure E.4, as μ_{CW,H_2O}=2.25 kgCW/kg and μ_{LPS,H_2O}=188.37 MJ/ton, respectively. Thus, the total amount of steam is μ_{LPS,H_2O}=400.51 MJ/ton.

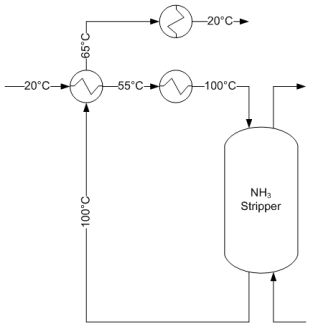


Figure E.4. Scheme of the ammonia stripper.

E.5.2 Air stripping

Ammonia can be removed with air stripping. The amount of air needed for stripping is calculated as for steam, resulting in μ_{N_2,H_2O}=101.92 kg_{Air}/ton.

E.6 Chemical oxidation

Chemical oxidation is used for the removal of organic compounds which are refractory, toxic, or inhibitory to microbial growth, and therefore cannot be removed through biological treatment. Chemical oxidation is characterized by extremely low selectivity, consequently, due to the existence of many side reactions, the efficiency of removal of a specific target compound is low.
For this reason, these processes are economically feasible only when a low enough concentration of organic contaminants are present in the wastewater. For these reasons, chemical oxidation is usually employed before the biological units or sometimes after them to further reduce the pollutant content.
Commonly used oxidants include ozone, permanganate, chlorine, chlorine dioxide, and ferrate. Oxidation is often done in combination with a catalyst, which include simple pH adjustment, UV light, transition metal cations, enzymes and a variety of proprietary catalysts. Standard mixed reactors with contact times between several minutes to an hour are used (Perry and Green, 2008). While the total COD decreases, the BOD and the soluble fractions increase due to the conversion from long chain biologically refractory organics to biodegradable compounds. The TOC does not decrease unless the compounds are completely oxidized. However, this is neglected, and the oxidation is considered only to target specific compounds.
In refinery wastewater, cyanides are often removed by oxidation. Thus, the oxidation of cyanides is considered here.

E.6.1 Cyanide oxidation

Cyanides are oxidized with hypochlorite (OCl^-). The oxidation of cyanide by chlorine proceeds through several reactions that are highly pH-dependent. Normally, a

two-steps process is employed: in the first step, cyanide react with hypochlorite to form cyanogen chloride which is subsequently hydrolyzed to form cyanate (pH 11.5); in the second step, cyanate is oxidized to form bicarbonates and nitrogen gas (pH 8.0 to 8.5). The overall reaction is:

$$5OCl^- + 2CN^- \rightarrow 5Cl^- + 2HCO_3^- + N_2 \tag{E.17}$$

If chlorine is used as a source of hypochlorite, the following dissociation reaction must be considered:

$$H_2O + H_2O \rightarrow OCl^- + Cl^- + 2H^+ \tag{E.18}$$

In reality, a large number of different reations exist, but for the sake of simplicity, only reaction E.17 (and E.18) are considered in this study.

Removal efficiency

Theoretically, any removal efficiency can be achieved depending on the amount of oxidant fed. It is reasonable to assume that a conversion of $\theta_{CN^-,E.17} = 0.99$ can be easily achieved.

Utility consumption

The utility consumption accounts for:

- Electricity for pumping.
- Oxidant.

The specific power consumption is estimated to be around 3.37 MJ/ton, based on the data reported by Environmental Protection Agency (1976) for a chlorination facility. The dosages of chlorine depend heavily on the identity of the metals complexed with the cyanide ion and the presence of other background constituents in the wastewater. Cyanide wastewaters containing nickel, silver or iron are difficult to treat by alkaline chlorination because of the slow reaction rate of these chemical processes (Eckenfelder *et al.*, 2009). Usually the amount of oxidant needed to achieve a certain performance is tuned experimentally for each wastewater. For simplicity, it is assumed that a 50% excess is supplied with respect to the stochiometric amount:

$$\mu_{H_2O,CN^-} = 1.5 \cdot \frac{\nu_{OCl^-,E.17} \cdot MW_{H_2O}}{\nu_{CN^-,E.17} \cdot MW_{CN^-}} \tag{E.19}$$

As a result, a specific consumption value μ_{H_2O,CN^-}=20.19 kg_{H_2O}/kg_{CN^-} is calculated.

E.7 Hydrothermal processes

Wastewater treatment steps performed at elevated temperatures and pressures are grouped as hydrotermal processes.

Although both Wet air oxidation (WAO) and hydrothermal hydrolysis and supercritical water oxidation (SCWO) have been proposed for the treatment of refinery wastewater, to date SCWO has not shown to be commercially viable, and no industrial application of this technology is reported (Eckenfelder *et al.*, 2009). Consequently, only WAO is considered in this study.

E.7.1 Wet air oxidation (WAO)

Wet air oxidation (WAO) is a treatment suitable for streams which are too diluted to incinerate and too concentrated for biological treatment. WAO consists of the oxidation of organic and inorganic substances in an aqueous solution with oxygen or air at elevated temperatures and pressures. Typical conditions for wet oxidation range from 180C to 315 C and 20 to 150 bar; residence times may range from 15 to 120 min, and the COD removal may typically be from 75 to 90% (Luck, 1999). Insoluble organic matter is converted to simpler soluble organic compounds which are in turn oxidized and eventually converted to carbon dioxide and water. Inorganics are partially oxidized to NH_4^+, N_2, SO_3^- or completely oxidized to NO_3^-, SO_4^{2-} or PO_4^{3-}.

Because of the elevated temperature and pressure at which the reactor is operated, considerable amount of power and energy are required for pumping the water and compressing the air, as well as for preheating purposes. The feed temperature is adjusted such that the exothermic heat of reaction raises the mixture temperature to the operating temperature, and the preheating can be done using the treated effluent. Different heat integration schemes have been proposed to limit the energy consumption associated to WAP. The first alternative consists of the production of low-pressure steam through the cooling of the WAO effluent. Alternatively, the heat can be used to preheat the wastewater to the operating conditions. Non-condensible gases are then separated from the effluent prior to further processing.

In most of the industrial applications, WAO is not used as a complete treatment method, but as a pretreatment aiming at reducing the toxicity of the water, and at reducing the COD level so that biological treatment becomes applicable. Consequently, milder operating conditions can be used, at the advantage of reduced operating and capital costs (Mishra *et al.*, 1995). Thus, in the superstructure, the WAO unit is positioned as a first possible treatment to decrease the COD in the high COD source, i.e. caustics.

For the sake of simplicity, only complete oxidation is considered, both for COD and H_2S:

$$C_{18}H_{19}O_9N + 17.5O_2 \rightarrow 18CO_2 + 8H_2O + NH_3 \tag{E.20}$$

$$H_2S + 2O_2 \rightarrow 2H^+ + SO_4^{2-} \tag{E.21}$$

After the treatment, pressure is relieved and the gas species produced through reaction (CO_2) and unreacted air are separated as gas, while the other species produced by the oxidation reaction remain dissolved in the water stream.

Removal efficiency

The conversion of the two reaction is assumed to be an average of the data reported in Luck (1999): $\theta_{C,E.20}, \theta_{H_2S,E.21}$=0.825.

Utility consumption

The utility consumption accounts for:

- Air for oxidation.
- Electricity for water pumping and air compression.
- Heating medium to raise the reactor temperature.
- Cooling medium.

In order to compute the utility requirements, average operating conditions of 247.5C and 85 atm are chosen. The necessary amount of oxygen required for COD oxidation is:

$$\mu_{O_2,C} = \theta_{C,E.20} \frac{\nu_{O_2,E.20} \cdot MW_{O_2}}{\nu_{C,E.20} \cdot MW_C} \tag{E.22}$$

If oxygen is supplied along with air, as it is often the case, also nitrogen is introduced as:

$$\mu_{N_2,C} = \frac{0.77}{0.23} \cdot \mu_{O_2,C} \tag{E.23}$$

In numbers, $\mu_{O_2,C}$=1.18 kg_{O_2}/kg_C and $\mu_{O_2,C}$=3.94 kg_{N_2}/kg_C. The total kg of air per kg of COD is 5.11 kg of air per kg of C.
Similarly, for oxidation of H_2S:

$$\mu_{O_2,H_2S} = \theta_{H_2S,E.21} \frac{\nu_{O_2,E.21} \cdot MW_{O_2}}{\nu_{H_2S,E.21} \cdot MW_{H_2S}} \tag{E.24}$$

$$\mu_{N_2,H_2S} = \frac{0.77}{0.23} \cdot \mu_{O_2,H_2S} \tag{E.25}$$

In numbers, μ_{O_2,H_2S}=1.55 kg_{O_2}/kg_{H_2S} and μ_{O_2,H_2S}=5.20 kg_{N_2}/kg_{H_2S}. The total kg of air per kg of H_2S is 6.75 kg of air per kg of H_2S.

The energy necessary to pump the water is calculated, for the given operating pressure, as μ^{el,H_2O} = 18.66 MJ/ton.
Similarly, the power required for the compression of air is calculated as (Biegler *et al.*, 1997):

$$w_C = \frac{\gamma}{\gamma - 1} \cdot R \cdot (T_{out} - T_{in}) \tag{E.26}$$

T_{in} being the initial gas temperature (20C) and T_{out} the final gas temperature; for an ideal system, γ=1.4 and if the expansion is assumed ideal and isotropic:

$$T_{out} = T_{in} (\frac{P_{out}}{P_{in}})^{\frac{\gamma-1}{\gamma}} \tag{E.27}$$

With the given conditions, T_{out}=765.71C and w_C=63,299.66 MJ per ton of air, which considering the air/COD and the air/H_2S ratios equals to 323.54 MJ/kg_C and 427.39 MJ/kg_{H_2S}.
Preheating of the incoming wastewater is performed using the hot outlet reactor stream. When the influent COD is above 20,000 mg/L, the exothermic reaction makes the process self-sustaining, and no auxiliary fuel requirement is needed (Mishra *et al.*, 1995); otherwise heating supply is needed. Here it is assumed that the utilities for heating and cooling can be neglected due to integration; the validity of this assumption need to be checked when the results are analysed.

E.8 Chemical precipitation

Chemical precipitation is a process employing chemicals to promote the precipitation of soluble components. In the case of wastewater treatment, it is adopted for metals, as well as for phosphorous removal. In this study, both phosphorous and metal precipitations are considered. If both options are selected, the possibility of fusing them in order to reduce the capital cost should be evaluated.

E.8.1 Phosphorous precipitation

Phosphorous precipitation is promoted by lime or alum. As an alternative, polymers have been also successfully employed. Phosphorous precipitation can be performed in various points in the wastewater treatment plant - before, during or after the biological treatment. However, due to the conversion of phosphorous to orthophosphorours (easier to precipitate) occurring in biological treatment, the precipitation is located after the biological treatment (Tchobanoglous *et al.*, 2003). In this study, phosphorous precipitation with lime is considered, a treatment suitable for wastewater with low alkalinity and high and variable phosphorous content (Eckenfelder *et al.*, 2009). Lime precipitation is governed by these reactions:

$$CaO + H_2O \rightarrow Ca^{2+} + 2OH- \tag{E.28}$$

if sufficient lime is mixed to increase the pH above 11, the precipitation of soluble phosphorous as calcium phosphate occurs:

$$10Ca^{2+} + 6PO_4^{3-} + 2OH- \rightarrow Ca_{10}(PO_4)_6(OH)_2 \tag{E.29}$$

CO_2 is then added to remove the excess calcium through precipitation as carbonate:

$$Ca^{2+} + 2OH - + CO_2 \rightarrow CaCO_3 + H_2O \tag{E.30}$$

Furthermore, the precipitation of calcium carbonate favour TSS removal, by acting as a coagulant.
Thermal regeneration at 980 C can be used to recover lime from the calcium carbonate contained in the sludge:

$$CaCO_3 \rightarrow CaO + CO_2 \tag{E.31}$$

In this study, lime regeneration is not considered, and the process is defined according to the simplified scheme shown in figure E.5. Consequently, process inputs are constituted by lime and CO_2.

Removal efficiency

The conversion of the reaction resulting in PO_4^{3-} removal is assumed $\theta_{PO_4^{3-}}$=0.90, the conversion of Ca^{2+} in reaction E.29 is $\theta_{Ca^{2+},E.29}$=0.60, while the rest is converted in E.30, so $\theta_{Ca^{2+},E.30}$=0.40. A perfect removal of the solids formed is considered, while a TSS removal of 80.0% is assumed.

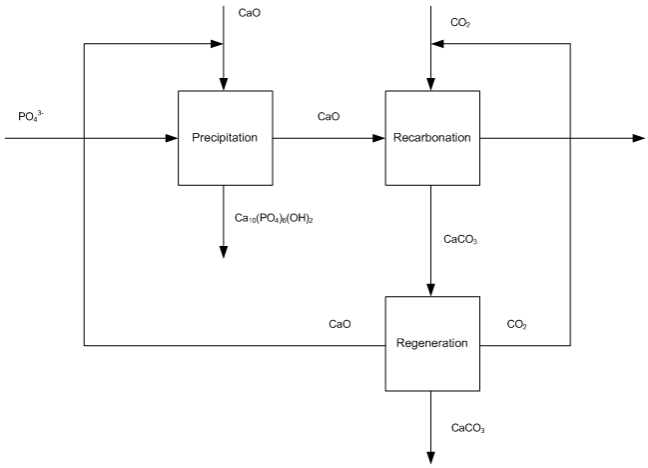


Figure E.5. Scheme of the lime precipitation process.

Utility consumption

The utility consumption accounts for:

- Electricity to run the equipment.
- Lime for precipitation.
- Make-up CO_2.

Since regeneration is not included, the fuel for the furnace is not considered.
The power consumption is estimated on the basis of what reported by the Environmental Protection Agency (1976) for a two-stage lime treatment and the average value of 5.74 MJ/ton is used.
The amount of lime to be fed varies greatly with the feed wastewater content and its alkalinity (Tchobanoglous *et al.*, 2003). For this case study, the dosage has been estimated assuming a 50% stoichiometric excess:

$$\mu_{CaO,PO_4^{3-}} = 1.5 \cdot \frac{\nu_{Ca^{2+},E.29} \cdot MW_{CaO}}{\nu_{PO_4^{3-},E.29} \cdot MW_{PO_4^{3-}}} \tag{E.32}$$

While carbon dioxide is fed in stoichiometric amount:

$$\mu_{CO_2,PO_4^{3-}} = \mu_{CaO,PO_4^{3-}} \cdot \theta_{Ca^{2+},E.30} \cdot \frac{MW_{CO_2}}{MW_{CaO}} \tag{E.33}$$

In numbers, $\mu_{CaO,PO_4^{3-}}$=1.47 $kg_{CaO}/kg_{PO_4^{3-}}$ and $\mu_{CO_2,PO_4^{3-}}$=0.46 $kg_{CO_2}/kg_{PO_4^{3-}}$.

E.8.2 Metal precipitation

Hydroxide ($OH-$), sulfide (S^{2-}), carbonates (CO_3^{2-}) and lime are commonly used chemicals for metals precipitation. Here, lime precipitation is considered. The

general reaction is:

$$Me^{n+} + nOH- \rightarrow Me(OH)_n \tag{E.34}$$

The reactions of metal precipitation are governed by equilibrium, which is influenced by the metal concentration and the pH of the solution. For simplicity, it is assumed that an equilibrium condition is obtained, which allows for precipitation of 95% of the metal in solution. One of the contaminant of major concern in refinery wastewater is hexavalent chromium Cr^{6+}; such a metal must first be reduced to the trivalent state (Cr^{3+}) using ferrous sulfate, and then precipitated with lime, according to reactions:

$$FeSO_4 \rightarrow Fe^{2+} + SO_4^{2-} \tag{E.35}$$

$$Cr^{6+} + 3Fe^{2+} \rightarrow Cr^{3+} + 3Fe^{3+} \tag{E.36}$$

This reaction occurs rapidly at pH levels below 3.0; acid must therefore be added for pH adjustment.
Then, Cr^{3+} is precipitated:

$$Cr^{3+} + 3OH- \rightarrow Cr(OH)_3 \tag{E.37}$$

The use of ferrous sulfate as a reducing agent has the disadvantage that a contaminating sludge of $Fe(OH)_3$ is formed:

$$Fe^{3+} + 3OH- \rightarrow Fe(OH)_3 \tag{E.38}$$

Consequently, the overall reaction is:

$$Cr^{6+} + 3Fe^{2+} + 12OH- \rightarrow Cr(OH)_3 + 3Fe(OH)_3 \tag{E.39}$$

In this study, two different treatment intervals are considered: the first one operates the generic metal removal according to reaction E.34, while the second is designed for removal of Cr^{6+} through reaction E.39.

Removal efficiency

Metal removal efficiency is dependent on the solubility equilibrium, which is influenced by composition, pH and temperature. As a simplification, a conversion of $\theta_{Cr^{6+},E.39}$=0.95 is assumed for reaction E.39. The solid material formed is completely removed. 80% TSS removal is assumed in the clarification.

Utility consumption

The utility requirement is:

- Electricity for pumping
- Lime
- Reducing agent $FeSO_4$ (only for Cr^{6+}).

The power consumption is assumed to be equal to lime precipitation.
The stoichiometric amount of lime to be fed for the unit mass of each metal to precipitate is calculated as:

$$\mu_{CaO,Me^{n+}} = n_{Me^{n+}} \cdot \frac{MW_{CaO}}{MW_{Me^{n+}}} \tag{E.40}$$

where $n_{Me^{n+}}$ is the valence of the metal. In case of Cr^{6+}:

$$\mu_{CaO,Cr^{6+}} = \frac{\nu_{OH-,E.39} \cdot MW_{CaO}}{\nu_{Cr^{6+},E.39} \cdot MW_{Cr^{6+}}} \tag{E.41}$$

In this latter case, $\mu_{CaO,Cr^{6+}}$=6.46 $\mathrm{kg}_{CaO}/\mathrm{kg}_{Cr^{6+}}$. With respect to $FeSO_4$, an excess of 150% is reported in the literature as requirement to achieve the desired conversion (Eckenfelder *et al.*, 2009):

$$\mu_{FeSO_4,Cr^{6+}} = 2.5\frac{\nu_{Fe^{2+},E.39} \cdot MW_{FeSO_4}}{MW_{Cr^{6+}}} \tag{E.42}$$

Which results in $\mu_{FeSO_4,Cr^{6+}}$=21.92 $kg_{FeSO_4}/kg_{Cr^{6+}}$.

E.9 Electrostatic separation

Ion exchange and the electrodialysis are the 2 treatment processes based on electrostatic forces considered in this case study.

E.9.1 Ion exchange (IE)

Ion exchange is a process in which ions of a given species are displaced from an insoluble exchange material by ions of a different species in solution. Ion exchange can be based on two types of resins: cation exchangers, which release hydrogen cations, or anion exchangers, which release hydroxyl anions. Ion exchange has been used for removal of total dissolved solids, heavy metals, and nitrogen. A wide spread application is water softening. Furthermore IE can be employed for the removal remove ammonium, and nitrate. In some cases IE can be used to recover valuable components (i.e. metals) from wastewater.

For a cationic resin, the IE process is:

$$nR^-A^+ + B^{n+} \leftrightharpoons R_n^-B^{n+} + nA^+ \tag{E.43}$$

where R^- is the anionic group in the resin matrix, B^{n+} is the cation to be removed, and A^+ is cation released in solution, which is typically H^+. The regeneration is done with H_2SO_4 as:

$$R_n^-B^{n+} + \frac{n}{2}H_2SO_4 \leftrightharpoons nR^-H^+ + B^{n+} + \frac{n}{2}SO_4^{2-} \tag{E.44}$$

For an anionic resin instead:

$$nR^+A^- + B^{n-} \leftrightharpoons R_n^+B^{n-} + nA^- \tag{E.45}$$

where R^+ is the cationic group in the resin, B^{n-} is the anion to be removed, and A^- is anion released in solution, which is typically $OH-$. Regeneration is done with $NaOH$:

$$R_n^+B^{n-} + nNaOH \leftrightharpoons nR^+OH - + n \tag{E.46}$$

Removal efficiency

A 95% removal efficiency is assumed for all ions.

Utility consumption

The utility consumption accounts for:

- Electricity for water pumping.
- Chemicals for regeneration ($NaOH$ or H_2SO_4).

Given the similarity between the two set-ups and the lack of specific data for IE, the power consumption is assumed to be equal to the one of a GAC adsorption bed. The consumption of chemicals for regeneration can be calculated by the stoichiometry of the regeneration reactions reported above.
For cation exchange, therefore, $n/2$ moles of H_2SO_4 are consumed for each mole of cation B^{n+} removed:

$$\mu_{H_2SO_4,B^{n+}} = \frac{n}{2} \cdot \frac{MW_{H_2SO_4}}{MW_{B^{n+}} \cdot wt} \tag{E.47}$$

where n is the valence of the ion to remove and wt is the weight purity of H_2SO_4.
For anionic exchange, similarly, n moles of $NaOH$ are required per mole of cation B^{n-} removed:

$$\mu_{NaOH,B^{n-}} = n \cdot \frac{MW_{NaOH}}{MW_{B^{n-}} \cdot wt} \tag{E.48}$$

where n is the valence of the ion to remove and wt is the weight purity of $NaOH$.

E.9.2 Electrodialysis (ED)

Ionic components of a solution are separated through the use of semipermeable ion-selective membrane thanks to application of an electric potential between two electrodes. ED is based on an electric current, which causes a migration of the ions. Usual retention times are between 10 and 20 days. A typical configuration consists of a treatment train composed by adsorption on active carbon followed by ED, in order to avoid salts precipitation on the electrodialysis membrane surface and clogging of the membrane by residual colloidal organic matter.

Removal efficiency

A product recovery between 75% and 85% is reported in Tchobanoglous *et al.* (2003), around 20% of water is wasted in the retentate. Ion removal is assumed to be 50%.

Utility consumption

Utilities account for:

- Electricity for pumping and membrane backwashing
- Chemicals for backwashing and washing

Average power consumption for ED is 9.5 kWh/m^3, which equals to μ_{EL,H_2O}=34.20 MJ/ton (Tchobanoglous *et al.*, 2003) . Beckwashing and washing is assumed to be equal to what decribed in the next section for membrane processing.

E.10 Membrane filtration

Membrane treatments employ a membrane, which is able to separate some of the contaminants, such as suspended and dissolved solids, colloids, metals or other ions. The feed is pressurized, and the permeate is collected at atmospheric pressure, creating a pressure difference which act as driving force across the membrane. While the contaminants build up on the membrane surface, filtration resistance is increased, until the membrane is backwashed, cleaned and subsequently replaced. Depending on driving force and sieve size, membrane filtration is categorized in different groups. Filtration technologies based on hydrostatic pressure difference are reported in E.2. Prior to filtration, the wastewater is often pretreated, with respect to pH regulation and, if necessary, disinfection to minimize or limit the bacteria growth on the membrane.

Table E.2. Membrane filtration processes relying on hydrostatic pressure difference (Tchobanoglous *et al.*, 2003).

Process	Pore size	Permeate	Constituents removed
Microfiltration	Macropores (≥50 nm)	Water, dissolved solutes	TSS
Ultrafiltration	Mesopores (2-50 nm)	Water, small molecules	Macromolecules, colloids
Nanofiltration	Micropores (≤2 nm)	Water, very small molecules, ionic solutes	Small molecules, some hardness
Reverse osmosis	Dense (≤2 nm)	Water, very small molecules, ionic solutes	Very small molecules, hardness, sulfates, nitrate, sodium, other ions

Microfiltration (MF) and Ultrafiltration (UF) membranes are commonly made of acrylonitrile, nylon, and polytetrafluoroethylene and have a lifetime of 5 years (Tchobanoglous *et al.*, 2003). Suspended solids removal and bacterial load reduction are commonly performed by MF (Tchobanoglous *et al.*, 2003).
Nanofiltration (NF) is used to remove selected dissolved constituents in wastewater such as metallic ions, inorganics, organics, bacteria and viruses, through membranes produced from polyamide or cellulose acetate (Tchobanoglous *et al.*, 2003).
Reverse osmosis (RO), finally, employs a pressure difference greater than the osmotic pressure in order to forces a water movement against the natural osmotic gradient (i.e., from concentrated solution to diluted solution). NF and RO cannot tolerate a too high contaminant load, therefore MF and UF is often used as pretreatment, in order to remove colloidal matter which would cause rapid membrane fouling.
In this study, MF and UF have been lumped in the same interval, as done for NF and RO.
The retentate separated by filtration has been treated as a waste, to which a waste

handling cost is associated.

Removal efficiency

Although the performances of these membrane is known for being highly site- and wastewater-specific, for the sake of this case study the typical performances reported in Tchobanoglous *et al.* (2003) for municipal wastewater are assumed. These data are reported in table E.3.

Utility consumption

Utility consumption accounts for:

- Electricity for pumping and backwashing.
- Chemicals for backwashing and washing.
- Pretreatment chemicals - acid or basic solution for pH regulation.

In table E.3, typical power consumption data are reported (Tchobanoglous *et al.*, 2003).

Table E.3. Power consumption and product recovery of various membrane technologies - adapted from Tchobanoglous *et al.* (2003).

Technology	MJ/ton	r
Microfiltration	1.44	0.96
Ultrafiltration	10.80	0.75
Nanofiltration	19.08	0.83
Reverse Osmosis	51.12	0.78

The cleaning protocol depends on the contaminant which is causing the fouling, and in general acids are used for mineral deposits, while alkaline solutions are employed for organic fouling. A variety of agents can be employed for the chemical cleaning of membranes, including detergents, acids, bases, oxidizing agents, sequestering agents, and enzymes; where the membrane material is not sensitive to chlorine, it can be employed in doses ranging from 2 to 2,000 mg/L (American Water Works Association, 2007). Based on the information reported in Eckenfelder *et al.* (2009), in this work it is assumed that 1.25% of the permeate water is used as cleaning solution every 48 hours. This resulted in a cleaning utility consumption of $\mu_{H_2O,H_2O} = 4.45 \cdot 10^{-4}$ kgH_2O/ton for MF/UF and $\mu_{H_2O,H_2O} = 4.17 \cdot 10^{-4}$ kgH_2O/ton for NF/RO.
It is thus assumed that the difference in the utility cost is associated to H_2SO_4 for pH pre-regulation and to $NaOH$ for pH post-regulation in equal measure. With respect to chemicals for pH control, a H_2SO_4 and $NaOH$ consumption of $\mu_{H2SO4,H2O}$=0.17 kg$_H2SO4$/kg$_{H_2O}$ and $\mu_{NaOH,H2O}$=0.02 kg$_NaOH$/kg$_{H2O}$ is assumed, based on the values reported by Park *et al.* (2010).

References

Alex, J.; Benedetti, L.; Copp, J.; Gernaey, K.; Jeppsson, U.; Nopens, I.; Pons, M.; Rosen, C.; Steyer, J. and Vanrolleghem, P. (2008). Benchmark Simulation Model no. 1 (BSM1). Technical report, The IWA Taskgroup on Benchmarking of Control Strategies for WWTPs.

American Water Works Association (2007). *Reverse osmosis and nanofiltration*. Manual of water supply practices. Denver.

Arena, B. and Buchan, M. (2006). Water and the refinery. In *AIChE Chicago Symposium*.

Barnicki, S. D. and Fair, J. R. (1990). Separation system synthesis: A knowledge-based approach. 1. Liquid mixture separations. *Industrial and Engineering Chemistry Research*, **29**, 421–432.

Barnicki, S. D. and Siirola, J. J. (2004). Process synthesis prospective. *Computers & Chemical Engineering*, **28**, 441–446.

Bek-Pedersen, E. and Gani, R. (2004). Design and synthesis of distillation systems using a driving-force- based approach. *Chemical Engineering and Processing: Process Intensification*, **43**(3), 251–262.

Bergamini, M. L.; Aguirre, P. and Grossmann, I. (2005). Logic-based outer approximation for globally optimal synthesis of process networks. *Computers & Chemical Engineering*, **29**(9), 1914–1933.

Bergamini, M. L.; Grossmann, I.; Scenna, N. and Aguirre, P. (2008). An improved piecewise outer-approximation algorithm for the global optimization of MINLP models involving concave and bilinear terms. *Computers & Chemical Engineering*, **32**(3), 477–493.

Biegler, L. T.; Grossmann, I. E. and Westerberg, A. W. (1997). *Systematic Methods of Chemical Process Design*. Prentice Hall Internatonal Series.

Birge, J. K. and Louveaux, F. (1999). *Introduction to Stochastic Programming (Springer Series in Operations Research and Financial Engineering)*. Springer, New York. ISBN 0-387-98217-5.

Bogataj, M. and Kravanja, Z. (2012). An alternative strategy for global optimization of heat exchanger networks. *Applied Thermal Engineering*, **43**(0), 75–90.

Bogataj, M; Bagajewicz, M. (2008). Synthesis of non-isothermal heat integrated water networks in chemical processes. *Computers & Chemical Engineering*, **32**(12), 3130–3142.

Bruggemann, S.; Bauer, N.; Fuchs, E.; Polt, A.; Wagner, B. and Wulkow, M. (2008). *Support of strategic business decisions at BASF's largest integrated production site based on site-wide Verbund simulation*, volume 25 of *Computer Aided Chemical Engineering*, pages 925–930. Elsevier. ISBN 1570-7946.

Brunet, J. C. and Liu, Y. (1993). Studies in chemical process design and synthesis An expert system for solvent-based separation process synthesis. *Industrial & Engineering Chemistry Research*, **32**, 315–334.

Carvalho, A.; Gani, R. and Matos, H. (2008). Design of sustainable chemical processes: Systematic retrofit analysis generation and evaluation of alternatives. *Process Safety and Environmental Protection*, **86**(5), 328–346.

Carvalho, A.; Matos, H. A. and Gani, R. (2013). SustainPro-A tool for systematic process analysis, generation and evaluation of sustainable design alternatives. *Computers and Chemical Engineering*, **50**, 8–27.

Chicago Board of Trade (1998). *Commodity trading manual*. Fitzroy Dearborn.

Clark, K.B.; Wheelwright, S. (1993). *Managing new product and process development*. Free Press.

Dominguez, L. and Pistikopoulos, E. (2010). Multiparametric programming based algorithms for pure integer and mixed-integer bilevel programming problems. *Computers*, **34**, 2097–2106.

Douglas, J. (1988). *Conceptual Design of Chemical Processes*. McGraw-Hill, New York.

Dua, V. and Pistikopoulos, E. N. (1998). Optimization Techniques for Process Synthesis and Material Design Under Uncertainty. *Chemical Engineering Research and Design*, **76**(3), 408–416.

Duran, M. and Grossmann, I. (1986a). An outer approximation algorithm for a class of mixed integer nonlinear PROGRAMS. *Mathematical Programming*, **36**, 307–339.

Duran, M. and Grossmann, I. (1986b). A mixed integer nonlinear programming algorithm for process synthesis. *Aiche Journal*, **32**, 592–606.

Eckenfelder, W. W.; Ford, D. L. and Englande, A. J. (2009). *Industrial Water Quality (4th Edition)*. McGraw-Hill.

Environmental Protection Agency (1976). Areawide Assessment Procedures Manual: Volume III.

Erdirik-Dogan, M.; Grossmann, I. E. and Wassick, J. (2007). A Bi-level decomposition scheme for the integration of planning and scheduling in parallel multiproduct batch reactors. *Computer Aided Chemical Engineering*, **Volume 24**, 625–630.

European Commission (2011). on the Catalogue of feed materials.

European Commission Integrated Pollution Prevention and Control (2003). Reference Document on Best Available Techniques in Common Waste Water and Waste Gas Treatment / Management Systems in the Chemical Sector.

Fiacco, A. and Bank, B. (1984). *Sensitivity, stability and parametric analysis*. North-Holland, Amsterdam.

Floudas, C. A. (1995). *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*, volume 2011. Oxford University Press, New York. id: 1.

GAMS Development Corporation (2011). *GAMS GDX facilities and tools*. GAMS Development Corporation, Washington DC, USA.

General Electrics (2012). Handbook of Industrial Wastewater Treatment. Retrieved from http://www.gewater.com/handbook/index.jsp in March 2012.

Gleick, P. H. (2012). *The World's Water Volume 7: The Biennial Report on Freshwater Resources*. Island Press.

Grossmann, I. (2005). Enterprise-wide optimization: A new frontier in process systems engineering. *AICHE Journal*, **51**(7), 1846–1857.

Grossmann, I. E.; Wiswanathan, J.; Vecchietti, A. and Raman, R. (2011). *GAMS - The Solver Manual*, chapter DICOPT, pages 241–261. GAMS Development Corporation, Washington DC, USA.

Gupta, O. and Ravindran, V. (1985). Bnetworks bound experiments in convex nonlinear Interger programming. *Management Science*, **31**, 1533–1546.

Halim, I. and Srinivasan, R. (2002). Integrated decision support system for waste minimization analysis in chemical processes. *Environmental Science and Technology*, **36**(7), 1640–1648.

Hammond, E. G.; Johnson., L. A.; Su, C.; Wang, T. and White, P. (2006). *Baileys industrial oil and fat products. Soybean Oil*, chapter Soybean Oil, pages 577–653. John Wileys & Sons., Hoboken, NJ.

Helton, J. C. and Davis, F. J. (2003). Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering & System Safety*, **81**(1), 23–69.

Henze, M.; Grady, C.; Gujer, W.; Marais, G. and Matsuo, T. (1987). Activated Sludge Model No. 1. Technical report, IAWPRC, London, U.K.

Henze, M.; Harremoes, P.; Jansen, J.; Arvin, E.; of Environmental Science, I.; of Denmark, T. T. U.; IMT and DTU (1993a). *Waste water treatment*. Lyngby.

Henze, M.; Harremoes, P.; Jansen, J. and E., A. (1993b). *Wastewater treatment: Biological and chemical processes*. Springer-Verlag, Berlin, third edition.

Henze, M.; Harremoes, P.; Jansen, J.; E., A.; Gujer, W.; T., M. and M.C.M., V. L. (2002). Activated Sludge Models ASM1, ASM2, ASM2d and ASM3. Technical report, IWA.

Hostrup, M.; Gani, R.; Kravanja, Z.; Sorsak, A. and Grossmann, I. (2001). Integration of thermodynamic insights and MINLP optimization for the synthesis, design and analysis of process flowsheets. *Computers & Chemical Engineering*, **25**, 73–83.

IBM Corp. (2009). *IBM ILOG CPLEX V 12.1*. IBM Corp.

Ierapetritou, M. G.; Acevedo, J. and Pistikopoulos, E. N. (1996a). An optimization approach for process engineering problems under uncertainty. *Computers & Chemical Engineering*, **20**(67), 703–709.

Ierapetritou, M. G.; Pistikopoulos, E. N. and Floudas, C. A. (1996b). Operational planning under uncertainty. *Computers & Chemical Engineering*, **20**(12), 1499–1516.

Iman, R. L. and Conover, W. J. (1982). A distribution-free approach to inducing rank correlation among input variables. *Communications in Statistics - Simulation and Computation*, **11**(3), 311–334.

IPIECA (2010). Petroleum refining water/wastewater use and management. Operations Best Practice Series 2010.

Jaksland, C. A.; Gani, R. and Lien, K. M. (1995). Separation process design and synthesis based on thermodynamic insights. *Chemical Engineering Science*, **50**(3), 511–530.

Judd, S. and Judd, C. (2006). *The MBR Book: Principles and Applications of Membrane Bioreactors in Water and Wastewater Treatment*. Elsevier.

Karuppiah, R. and Grossmann, I. E. (2006). Global optimization for the synthesis of integrated water systems in chemical processes. *Computers & Chemical Engineering*, **30**(4), 650–673.

Karuppiah, R. and Grossmann, I. E. (2008). Global optimization of multiscenario mixed integer nonlinear programming models arising in the synthesis of integrated water networks under uncertainty. *Computers & Chemical Engineering*, **32**(12), 145–160.

Khor, C. S.; Foo, D. C. Y.; El-Halwagi, M.; Tan, R. R. and Shah, N. (2011). A Superstructure Optimization Approach for Membrane Separation-Based Water Regeneration Network Synthesis with Detailed Nonlinear Mechanistic Reverse Osmosis Model. *Industrial and Engineering Chemistry Research*, **50**(23), 13444–13456.

Khor, C. S.; Chachuat, B. and Shah, N. (2012). A superstructure optimization approach for water network synthesis with membrane separation-based regenerators. *Computers & Chemical Engineering*, **42**(0), 48–63.

King, D. G. and Barnes, F. (1972). Systematic Evolutionary Process Synthesis. *Industrial & Engineering Chemistry Process Design and Development*, (2), 271.

Kopanos, G. M.; Puigjaner, L. and Georgiadis, M. C. (2009). A Bi-Level Decomposition Methodology for Scheduling Batch Chemical Production Facilities. *Computer Aided Chemical Engineering*, **27**, 681–686.

Kurkkio, M.; Frishammar, J. and Lichtenthaler, U. (2011). Where process development begins: A multiple case study of front end activities in process firms. *Technovation*, **31**, 490–504.

Luck, F. (1999). Wet air oxidation: past, present and future. *Catalysis Today*, **53**(1), 81–91.

Lutze, P.; Babi, D. K.; Woodley, J. M. and Gani, R. (2013). Phenomena Based Methodology for Process Synthesis Incorporating Process Intensification. *Industrial & Engineering Chemistry Research*, pages –.

McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs - 1. Convex underestimating problems. *Math Program*, **10**(2), 147–175.

McGalliard, R. L. and Westerberg, A. W. (1972). Structural sensitivity analysis in design synthesis. *The Chemical Engineering Journal*, **4**(2), 127–138.

Meidl, J. A. (1997). Responding to changing conditions: how powdered activated carbon systems can provide the operational flexibility necessary to treat contaminated groundwater and industrial wastes. *Carbon*, **35**(9), 1207–1216.

Merlo, R.; Gerhardt, M. B.; Burlingham, F.; Casas, C. D. L.; Gill, E. and Flippin, T. H. (2011). Petroleum Refinery Stripped Sour Water Treatment Using the Activated Sludge Process. *Water Environment Research*, **83**(11), 2067–2078.

Mishra, V. S.; Mahajani, V. V. and Joshi, J. B. (1995). Wet air oxidation. *Industrial and Engineering Chemistry Research*, **34**(1), 2–48.

Park, C.; Park, P.; Mane, P. P.; Hyung, H.; Gandhi, V.; Kim, S. and Kim, J. (2010). Stochastic cost estimation approach for full-scale reverse osmosis desalination plants. *Journal of Membrane Science*, **364**(2), 52–64.

Parkash, S. (2003). *Refining Processes Handbook*. Gulf Professional Publishing.

Perry, R. H. and Green, D. W. (2008). *Perry's chemical engineers' handbook*. McGraw-Hill, New York.

Pistikopoulos, E. and Mazzuchi, T. (1990). A novel flexibility analysis approach for processes with stochastic parameters. *Computers*, **14**, 991–1000.

Pistikopoulos, E. N; Georgiadis, M. D. V. (2007). *Multi-Parametric Programming: Theory, Algorithms, and Applications*. Wiley.

Puckett, D. A. (2008). Water Management in Process Plants. University of Oklahoma, Department of Chemical, Biological, and Materials Engineering.

Qasim, S. R. (1985). *Wastewater treatment plants / Planning, design, and operation*. CBS Publishing Japan ltd, New York.

Quaglia, A.; Sarup, B.; Sin, G. and Gani, R. (2012). Integrated business and engineering framework for synthesis and design of enterprise-wide processing networks. *Computers & Chemical Engineering*, **38**, 213–223.

Quaglia, A.; Sarup, B.; Sin, G. and Gani, R. (2013a). A Systematic Framework for Enterprise-Wide Optimization: Synthesis and Design of Processing Networks Under Uncertainty. *Computers & Chemical Engineering*, **(in press)**.

Quaglia, A.; Sarup, B.; Sin, G. and Gani, R. (2013b). Design of a Generic and Flexible Data Structure for Efficient Formulation of Large Scale Network Problems. *Computer Aided Chemical Engineering*, **(in press)**.

Quaglia, A.; Pennati, A.; Bogataj, M.; Kravanja, Z.; Sin, G. and Gani, R. (submitted). Industrial process water treatment and reuse: A framework for synthesis and design. *Industrial & Engineering Chemistry Research*.

Raman, R. and Grossmann, I. E. (1991). Relation between MILP modelling and logical inference for chemical process synthesis. *Computers & Chemical Engineering*, **15**(2), 73–84.

Roels, J. A. (1983). *Energetics and kinetics in biotechnology*. Elsevier Biomedical Press, Amsterdam.

Rojas-Torres, M. G.; Ponce-Ortega, J. M.; Serna-Gonzlez, M.; Npoles-Rivera, F. and El-Halwagi, M. M. (2013). Synthesis of water networks involving temperature-based property operators and thermal effects. *Industrial and Engineering Chemistry Research*, **52**(1), 442–461.

Sahinidis, N. and Tawarmalani, M. (2011). *GAMS - The Solver Manual*, chapter BARON, pages 21–36. Gams Development Corporation, Washington DC, USA.

Schultz, T. E. (2005). Get the most out of API separator. *Chemical Engineering*, **112**(7), 38–42.

Sebastiani, E. G.; Snoeyink, V. L. and Angelotti, R. W. (1994). Thermal regeneration of spent and acid-washed GAC from the upper occoquan sewage authority. *Water Environment Research*, **66**(3), 199–205.

Shapiro, A. (2012). Minimax and risk averse multistage stochastic programming. *European Journal of Operational Research*, **219**(3), 719–726.

Siirola, J. J. (1996). Strategic process synthesis: Advances in the hierarchical approach. *Computers & Chemical Engineering*, **20**(0), 1637–1643.

Simon, H. A. (1947). *Administrative Behavior: a Study of Decision-Making Processes in Administrative Organization*. Macmillan, New York, iv edition.

Simon, H. A. (1972). *Theories of Bounded Rationality*. Decision and Organization. North Holland, Amsterdam.

Simon, H. A. (1973). Administrative Decision Making. *IEEE Eng. Manag. Rev.*, **1**(1), 60–66.

Simon, H. A. (1981). *The sciences of the artificial. Cambridge, Mass: MIT Press.*. MIT Press, Cambridge.

Sin, G.; Hulle, S. V.; Pauw, D. D.; van Griensven, A. and Vanrolleghem, P. (2005). A critical comparison of systematic calibration protocols for activated sludge models: A SWOT analysis. *Water Research*, **39**, 2459–2474.

Sin, G.; Lantz, A. E. and Gernaey, K. V. (2009). Sensitivity Analysis of Non-Linear Dynamic Models: Prioritizing Experimental Research. *Computer Aided Chemical Engineering*, **27**, 267–272.

Stephanopoulos, G. and Westerberg, A. W. (1976). Studies in process synthesis-II. Evolutionary synthesis of optimal process flowsheets. *Chemical Engineering Science*, **31**(3), 195–204.

Straub, D.A; Grossmann, I. (1993). Design optimization of stochastic flexibility. *Computers*, **17**, 339–354.

Swaney, R. and Grossmann, I. (1985a). An index for operational flexibility in chemical process design. Part I: formulation and theory. *Aiche Journal*, **31**, 621.

Swaney, R. and Grossmann, I. (1985b). An index for operational flexibility in chemical process design. Part I: computational algorithm. *Aiche Journal*, **31**, 621.

Takama, N.; Kuriyama, T.; Shiroko, K. and Umeda, T. (1980). Optimal water allocation in a petroleum refinery. *Computers & Chemical Engineering*, **4**(4), 251–258.

Tan, R. R.; Ng, D. K. S.; Foo, D. C. Y. and Aviso, K. B. (2009). A superstructure model for the synthesis of single-contaminant water networks with partitioning regenerators. *Process Safety and Environmental Protection*, **87**(3), 197–205.

Tawarmalani, M. and Sahinidis, N. V. (2005). A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, **103**(2), 225–249.

Tay, D.; Ng, D.; Kheireddine, H. and El-Halwagi, M. (2011). Synthesis of an integrated biorefinery via the C-H-O ternary diagram. *Clean Technologies and Environmental policy*, **13**, 567–579.

Tchobanoglous, G.; Burton, F. L.; Stensel, H. D. and Inc, M. (2003). *Wastewater engineering / Treatment and reuse*. McGraw-Hill, New York.

Teles, J. P.; Castro, P. M. and Matos, H. A. (2012). Global optimization of water networks design using multiparametric disaggregation. *Computers & Chemical Engineering*, **40**(0), 132–147.

Terrazas-Moreno, S. and Grossmann, I. E. (2011). A multiscale decomposition method for the optimal planning and scheduling of multi-site continuous multiproduct plants. *Chemical Engineering Science*, **66**, 4307–4318.

Thoenes, P. (2009). *Oilseeds, Oils and Meals*, volume 2011 of *Food Outlook - June 2009*, chapter 7/20/2011. FAO. id: 1.

Tran, F. T. and Chowdhury, A. K. M. M. (1992). Performance of RBC coupled to a polyurethane foam to biodegrade petroleum refinery wastewater. *Environmental Pollution*, **76**(1), 61–70.

Turkay, M. and Grossmann, I. (1996). Logic based MINLP algorithm for the optimal synthesis of process networks. *Computers & Chemical Engineering*, **20**, 959–978.

Viswanathan, J. and Grossmann, I. E. (1990). A combined penalty function and outer-approximation method for MINLP optimization. *Computers & Chemical Engineering*, **14**(7), 769–782.

Wang, L. K. (2010). *Flotation technology*. Humana Press, New York. Series: Handbook of environmental engineering; 12.

Westerlund, T.; Skrifvars, H. H. I. (1998). An extended cutting plane method for a class of non convex MINLP problems. *Computers & Chemical Engineering*, **22**, 357–365.

Wheelwright, S. C.;Clark, K. B. (1992). *Revolutionizing Product Development*. The Free Press, New York.

Wong, J. and Hung, Y. T. (2004). *Treatment of Oilfield and Refinery Wastes*. Handbook of Industrial and Hazardous Wastes Treatment. CRC Press.

World Bank (1999). Pollution Prevention and Abatement Handbook 1998 - Toward Cleaner Production. TY: GEN; Owner: Ale; Added to JabRef: 2012.04.17.

World Intellectual Property Organization (2004). Succesfull technology licensing. Geneva Switzerland.

Yee, T.; Grossmann, I. and Kravanja, Z. (1990). Simultaneous optimization models for heat integration -I. Area and energy targeting and modeling of multi-stream exchangers. *Computers & Chemical Engineering*, **14**, 1151–1164.

Zarooni, M. A. and Elshorbagy, W. (2006). Characterization and assessment of Al Ruwais refinery wastewater. *Journal of hazardous materials*, **136**(3), 398–405.

Zondervan, E.; Nawaz, M.; de Haan, A. B.; Woodley, J. M. and Gani, R. (2011). Optimal design of a multi-product biorefinery system. *Computers & Chemical Engineering*, **35**(9), 1752–1766.

This PhD-project was carried out at CAPEC, the Computer Aided Product-Process Engineering Center.
CAPEC is committed to research, to work in close collaboration with industry and to participate in educational activities. The research objectives of CAPEC are to develop computer-aided systems for product/process simulation, design, analysis and control/operation for chemical, petrochemical, pharmaceutical and biochemical industries. The dissemination of the research results of CAPEC is carried out in terms of computational tools, technology and application. Under computational tools, CAPEC is involved with mathematical models, numerical solvers, process/operation mathematical models, numerical solvers, process simulators, process/product synthesis/design toolbox, control toolbox, databases and many more. Under technology, CAPEC is involved with development of methodologies for synthesis/design of processes and products, analysis, control and operation of processes, strategies for modelling and simulation, solvent and chemical selection and design, pollution prevention and many more. Under application, CAPEC is actively involved with developing industrial case studies, tutorial case studies for education and training, technology transfer studies together with industrial companies, consulting and many more.

Further information about CAPEC can be found at www.capec.kt.dtu.dk.

Computer Aided Process Engineering Center
Department of Chemical and Biochemical Engineering
Technical University of Denmark
Søltofts Plads, Building 229
DK-2800 Kgs. Lyngby
Denmark

Phone: +45 4525 2800
Fax: +45 4525 4588
Web: www.capec.kt.dtu.dk

ISBN : 978-87-93054-08-0